Available online at www.sciencedirect.com
ScienceDirect
Theoretical Computer Science 381 (2007) 1–28
Theoretical Computer Science
www.elsevier.com/locate/tcs

Fundamental study

# From sandpiles to sand automata[☆]

Julien Cervelle[a], Enrico Formenti[b,*], Benoît Masson[b]

[a] *Institut Gaspard-Monge, Université de Marne-la-Vallée, 77454 Marne-la-Vallée Cedex 2, France*
[b] *Laboratoire I3S, Université de Nice-Sophia Antipolis, CNRS, 2000 Route des Lucioles, 06903 Sophia Antipolis, France*

Received 13 February 2006; received in revised form 19 March 2007; accepted 20 March 2007

Communicated by B. Durand

**Abstract**

We introduce a new model for simulating natural phenomena. We address several issues: topology, basic set properties like injectivity and surjectivity, reversibility, and decidability questions about a special kind of conservation law called grain conservation and ultimate periodicity.


*Keywords:* Sandpiles; Sand automata; Reversibility; Undecidability; Grain conservation; Ultimate periodicity

## 1. Introduction

Sandpiles are a paradigmatic example for systems ruled by self-organized criticality (SOC). SOC is a very common phenomenon observed in a huge variety of processes in physics, biology and computer science.

Basically, a SOC system reaches a "critical" state after some finite transient. Any perturbation of this critical state, no matter how small, generates a deep uncontrollable reorganization of the whole system. Then, after some other finite transient, the system reaches a new critical state and so on.

Sandpiles well illustrate this phenomenon. Indeed, consider dropping sand grains on a table, one by one. Little by little a sandpile starts growing. It will steepen until the slope at its edges reaches some critical value. Any further addition of grains will cause cascades of grains to topple down. Finally, after a spontaneous spatial redistribution of grains a new "stable" state is reached. Afterwards, the sandpile will start evolving towards a new critical edge and so on.

An interesting formal model for sandpiles has been introduced in [3–5]. It is based on a local interaction rule (see Section 3). The simplicity of the formalization contrasts with the complexity of the dynamical behavior. Indeed, it exhibits all the characteristics of a typical SOC system.

☆ Part of the results of this paper have been presented at STACS 2003 [J. Cervelle, E. Formenti, On sand automata, in: STACS 2003: 20th Annual Symposium on Theoretical Aspects of Computer Science, in: Lecture Notes in Computer Science, vol. 2607, Springer, 2003, pp. 642–653] and MFCS 2005 [J. Cervelle, E. Formenti, B. Masson, Basic properties for sand automata, in: Mathematical Foundations of Computer Science 2005, in: Lecture Notes in Computer Science, vol. 3618, Springer, 2005, pp. 192–211] conferences.

* Corresponding author. Tel.: +33 4 92 07 66 56.
*E-mail addresses:* julien.cervelle@univ-mlv.fr (J. Cervelle), enrico.formenti@i3s.unice.fr (E. Formenti), benoit.masson@i3s.unice.fr (B. Masson).

0304-3975/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2007.03.042

For these reasons it received a great deal of attention over the years [3,4,6–9]. Several variants have been proposed to study the integer partitions [10], the structure of the phase space [11,12,5], and how perturbations of local rules interact with this structure [13,14].

Most of the results in this context have been obtained by algebraic and combinatorial approaches exploiting the lattice structure of the phase space. The issue is that all these results cannot be easily generalized. For this reason, we started stepping to the classical discrete dynamical systems point of view [1,2]. The first step is to provide a suitable topology on sandpiles. Of course, some basic requirements like compactness and perfectness on the topology are necessary in order to ease the investigation.

In Section 2, we introduce a new metric on configurations (*i.e.* spatial distributions of sand grains); the induced topology is locally compact, perfect and totally disconnected. In this setting, a sandpile is nothing but a continuous function acting on configurations on the basis of a local interaction rule.

Sand automata generalize this notion (see Section 3). Their formal definition is similar to cellular automata with the supplementary constraint that modifications on a configuration should obey some consistency rule. For example, if a column contains a certain number of grains then, after the application of the local rule, it may contain a different amount of grains but there are no holes, *i.e.* grains are always as clustered as possible.

They can be characterized by a Hedlund-like theorem, a fundamental representation result which says that the class of sand automata is exactly the class of infiniteness conserving continuous functions commuting with the shift and the raising maps (see Section 4). Moreover, this theorem helps in proving that the inverse of a sand automaton is still a sand automaton.

This result (together with the new metric on configurations) makes sand automata a completely new model although there are many connections with cellular automata (see Section 5). This claim is well illustrated by the results of Section 6 where we study the relations between basic set properties like surjectivity and injectivity (compare for instance, with the similar results about cellular automata reported in [15]).

In the second part of the paper we address two decidability issues: grain conservation and ultimate periodicity. A system $\mathcal{S}$ is *grain conserving* if the total number of grains is conserved during the evolution of $\mathcal{S}$. In Section 7, we have proved that this property is decidable in any dimension.

Continuing the parallel with sandpiles: we know that these systems reach a fixed point after some finite transient which might depend on the number of grains and on their spatial distribution [10,13]. In a more general setting, one can wonder whether a given sand automaton reaches a periodic point after some finite transient, *i.e.* it is *ultimately periodic*. In Section 8, we prove that this problem is undecidable by reducing it to the halting problem of a two registers machine.

## 2. A topology on sandpiles

The dynamical systems approach often requires some topology on the space on which they act. In this section we introduce a metric topology on configurations and we study its main properties.

A *configuration* represents a set of sand grains, organized in piles and distributed all over a $d$-dimensional grid. Every point of the grid $\mathbb{Z}^d$ is associated with the *number of grains i.e.* an element of $\widetilde{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty, +\infty\}$. The value $-\infty$ represents a *sink* and $+\infty$ a *source* of sand grains. More formally, a configuration is an element of $\widetilde{\mathbb{Z}}^{\mathbb{Z}^d}$. Denote by $x_{i_1,\ldots,i_d}$ or $x_i$ the number of grains in the column of $x$ indexed by the vector $i = (i_1, \ldots, i_d)$. Denote $\mathfrak{C}$ the set of all configurations. Finally, for $u \in \widetilde{\mathbb{Z}}$, let $\mathfrak{C}_u$ be the set of configurations whose sand amount at position $(0, \ldots, 0)$ is $u$.

When the dimension $d$ is known without ambiguity we note 0 the vector $(0, \ldots, 0)$ of $\mathbb{Z}^d$ and for a vector $i \in \mathbb{Z}^d$, we will note $|i| = \max |i_l|$, $1 \leqslant l \leqslant d$, the infinite norm of vector $i$. In order to compare two vectors $i, j \in \mathbb{Z}^d$, denote $i \preccurlyeq j$ the fact that for all $k \in [\![1, d]\!]$, $i_k \leqslant j_k$. If $i \preccurlyeq j$ and $i \neq j$ then we write $i \prec j$.

Before formalizing the definition of the distance, we explain in a few sentences, using a metaphor, how two configurations can be compared.

Assume there is an operator who has a set of tools for measuring the heights of sandpiles in a configuration. Each tool has a measuring limit and can make measurements only on a finite range of sites.

Heights bigger than the tool limit are declared to be infinite. If the operator estimates that more precision is needed, he may decide to change the tool to a more powerful one.

The problem is how to measure the distance between two configurations. This is a difficult problem because we want to choose our metric so to obtain a (locally) compact space and, at the same time, not to lose the intuitiveness and adequacy of the sandpiles context.

We propose the following behavior for the operator when measuring the distance between two configurations $x$ and $y$. First of all, he chooses a reference point, the site of index 0 for instance. If the number of grains at index 0 are different, then the operator declares those configurations completely different, *i.e.* at distance 1. Otherwise, the configurations will be observed putting a measuring device of precision $r$, starting with $r = 1$ on top of the pile of the reference point. The height of this pile will be referred to as *the reference height*. From the reference point, for each of the sites $i$ which are near it *i.e.* $|i| \leqslant r$, the operator will note the difference of height between the sandpile at the reference and at site $i$. This difference is declared infinite if it is greater than $r$ (and therefore out of sight of the measuring device). If the current device is precise enough to point out a difference between $x$ and $y$, then the distance between $x$ and $y$ is $2^{-r}$. Otherwise the operator starts the process again using a more powerful measuring device *i.e.* of precision $r + 1$. The process continues until he can distinguish between $x$ and $y$.

Before giving the formal definition of the distance between configurations we need a few more notations.

For all $a, b \in \mathbb{Z}$ such that $a \leqslant b$, let $[\![a, b]\!] = \{a, a + 1, \ldots, b\}$ and $\widetilde{[\![a, b]\!]} = [\![a, b]\!] \cup \{+\infty, -\infty\}$. "Measuring devices" of precision $r \in \mathbb{N}$ and reference height $m \in \mathbb{Z}$ are nothing but functions from $\mathbb{Z}$ to $\widetilde{[\![-r, r]\!]}$ defined as follows

$$\beta_r^m(n) = \begin{cases} +\infty & \text{if } n > m + r, \\ -\infty & \text{if } n < m - r, \\ n - m & \text{otherwise.} \end{cases}$$

**Remark 1.** When the precision is increased from $r$ to $r + 1$, if the observed value is finite, it remains finite. However, if it was infinite, it can either remain infinite, or turn into $r$ or $-r$, depending on the sign of the infinity.

**Definition 2** (*Cylinder*). For any configuration $x \in \mathfrak{C}$, $r \in \mathbb{N} \setminus \{0\}$ and $i \in \mathbb{Z}^d$ let $d_r^i(x)$ be the matrix of dimension $d$, for $k \in [\![-r, r]\!]^d$,

$$d_r^i(x)_k = \begin{cases} x_i & \text{if } k = 0, \\ \beta_r^{x_i}(x_{i+k}) & \text{if } k \neq 0 \text{ and } |x_i| < \infty, \\ \beta_r^0(x_{i+k}) & \text{otherwise.} \end{cases}$$

These are the measures observed by the operator using the device $\beta_r$ and site $i$ as a reference point (see Fig. 1 for an example in dimension 1).

The matrix $w = d_r^i(x)$ is called *cylinder* of radius $r$. The height $x_i$ from which measures are taken is put at the center of the matrix $d_r^i(x)$. Moreover, it is called the *reference* point of the cylinder and is denoted $w_0$ (for any $w = d_r^i(x)$, $w_0 = x_i$).

Remark that when $x_i$ is infinite, a measuring device centered on the infinite column $x_i$ would not be able to distinguish finite values in the neighborhood. This is why in that case the reference point is set at height 0.

**Definition 3.** The *distance* between two configurations $x$ and $y$ is defined as $d(x, y) = 2^{-r}$, where $r$ is the least integer such that $d_r^0(x) \neq d_r^0(y)$.

**Proposition 4.** *The map $d$ is a distance.*

**Proof.** The facts that $d(x, y) = 0 \Leftrightarrow x = 0$ and that $d$ is symmetric are obvious. Finally, remark that $d$ is ultrametric, *i.e.* $d(x, z) \leqslant \max(d(x, y), d(y, z))$. Indeed, suppose that $r$ is the least integer such that $d_r^0(x) \neq d_r^0(y)$ and that $s$ is the least integer such that $d_s^0(y) \neq d_s^0(z)$. Let $t = \min(s, r)$. Then for all $u < t$, $d_u^0(x) = d_u^0(y) = d_u^0(z)$ and hence, $d(x, z) \leqslant 2^{-t} \leqslant \max(d(x, y), d(y, z))$. □

Using Remark 1, if $d(x, y) = 2^{-r}$, then for all integer $k$ greater than $r$ we have $d_k^0(x) \neq d_k^0(y)$.

In the following, the space $\mathfrak{C}$ is a topological space endowed with the topology induced by $d$. In this metric space, cylinders are a base of open sets for the topology. Given a cylinder $w$, the open ball of radius $2^{-r}$ induced by $w$ is $\left\{x \in \mathfrak{C}, d_r^0(x) = w\right\}$. It is denoted by $[w]_r$ and sometimes $[w]$. Note that the distance being ultrametric, every point $x$ such that $d_r^0(x) = w$ is at the center of the ball $[w]_r$. See Fig. 2 for an example of open ball.

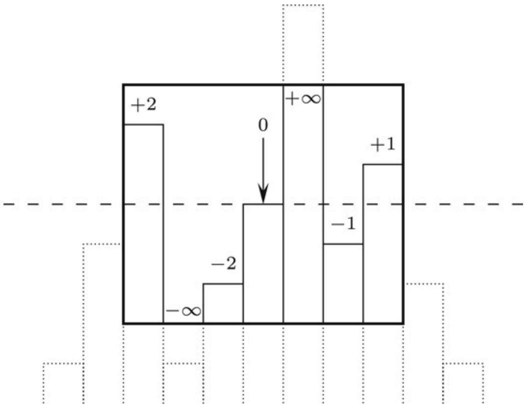



Fig. 1. Illustration of a "measuring device" of precision 3 in dimension 1. In this example $d_3^0(x) = (2 \quad -\infty \quad -2 \quad 0 \quad +\infty \quad -1 \quad 1)$ is the "cylinder" of radius 3.

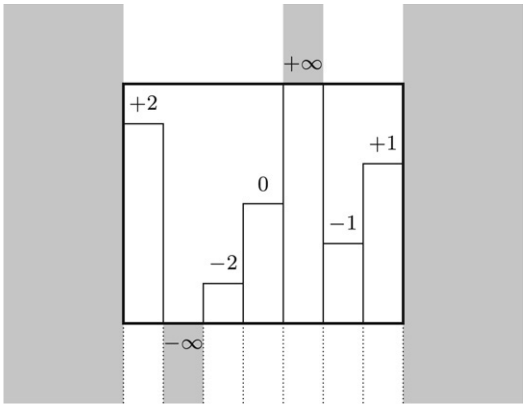



Fig. 2. Open ball of radius $2^{-4}$, centered on the one-dimensional cylinder $(2 \quad -\infty \quad -2 \quad 0 \quad +\infty \quad -1 \quad 1)$ from Fig. 1. The grayed areas represent the set of possible values for the elements of the ball, while the plain lines indicate the fixed columns.

In the remaining part of this section we investigate the properties of the topology induced by $d$. Most of these results are heavily used in the following and they are proved in [1]. Here, the proofs are generalized to any dimension.

**Proposition 5.** *The space $\mathfrak{C}$ is perfect (*i.e. *it has no isolated points).*

**Proof.** Choose an arbitrary configuration $x \in \mathfrak{C}$. For any $l \in \mathbb{N}$, build a configuration $x' \in \mathfrak{C}$, equals to $x$ except at site $\ell = (l+1, 0, \dots, 0)$, defined as follows

$$\forall j \in \mathbb{Z}^d \setminus \{\ell\},\ x'_j = x_j \quad \text{and} \quad x'_\ell = \begin{cases} 0 & \text{if } x_\ell \neq 0, \\ 1 & \text{otherwise.} \end{cases}$$

By definition, $0 < d(x, x') \leqslant 2^{-l-1}$. $\square$

Many classical results in discrete dynamical systems dynamics rely on the compactness of the space. Unfortunately, $\mathfrak{C}$ is not compact. In fact, it is easy to see that the sequence $(x^n)_{n\in\mathbb{N}}$, where $x_0^n = n$ and $x_i^n = 0$ for $i \neq 0$, has no converging subsequence. Corollary 8 proves that $\mathfrak{C}$ is at least locally compact. A central role in the proof of this result is played by the sets $\mathfrak{C}_u = \{x \in \mathfrak{C},\ x_0 = u\}$, for $u \in \widetilde{\mathbb{Z}}$. These sets are characterized in the following proposition.

**Proposition 6.** *For all $u \in \widetilde{\mathbb{Z}}$, the set $\mathfrak{C}_u$ is compact.*

**Proof.** We need to order the sites of a configuration from the center, non-decreasingly with respect to the infinite norm. To this extent, we choose a bijection $f$ from $\mathbb{N}$ to $\mathbb{Z}^d$ such that, if $i < j$, then $|f(i)| \leqslant |f(j)|$. Note that $f(0) = 0$. An example for such an $f$ is illustrated on Fig. 3.

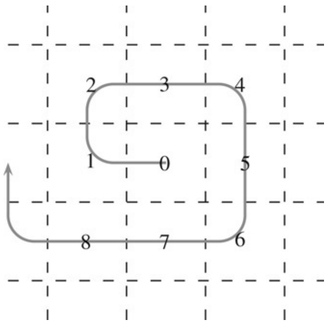



Fig. 3. Sample $f$ in dimension 2.

Consider an infinite set $E$ of configurations in $\mathfrak{C}_u$. We are going to build a configuration $y \in E$ such that for all $\varepsilon > 0$, there exists infinitely many $x \in E$ such that $d(x, y) \leqslant \varepsilon$. Let $y_0 = u$. We assign the values for $y$ at positions $f(1)$, $f(2)$, $f(3)$ *etc.* Let $U_0 = E$, we build $(U_i)_{i\in\mathbb{N}}$, a non-increasing sequence of infinite sets of configurations of $E$. For each value $i \neq 0$, consider the sequence $(x_{f(i)})_{x\in U_{i-1}}$; there are three possible cases:

(i) there is a value $k_i$ which occurs infinitely many times and we set $y_{f(i)} = k_i$, and let $U_i$ the infinite set of configurations $x$ of $U_{i-1}$ such that $x_{f(i)} = k_i$;
(ii) there exists a strictly increasing subsequence, set $y_{f(i)} = +\infty$, and let $U_i \subset U_{i-1}$ be an infinite set of configurations such that for any integer $l$, there are only finitely many $x \in U_i$ such that $x_{f(i)} < l$.
(iii) there exists a strictly decreasing subsequence, set $y_{f(i)} = -\infty$, and let $U_i \subset U_{i-1}$ be an infinite set of configurations such that for any integer $l$, there are only finitely many $x \in U_i$ such that $x_{f(i)} > l$.

Let us prove that $y$ has the required property. Let $l$ be a positive integer. We want to find infinitely many configurations $x \in E$ such that $d(x, y) \leqslant 2^{-l}$. Let $\ell \in \mathbb{Z}$ be such that $|f(\ell)| = l + 1$. Consider the set $U_\ell$. For all configurations $x \in U_\ell$, and for all $k$ such that $|k| \leqslant l$, either $y_k = x_k$ or $y_k = +\infty$ [resp. $-\infty$], and only finitely many configurations $x$ in $U_\ell$ are such that $x_k \leqslant l + x_0$ [resp. $x_k \geqslant -l - x_0$], which we can remove from $U_\ell$ keeping it infinite. For any of the remaining $x$ from $U_\ell$ we have that for all $k$ such that $|k| \leqslant l$, $y_k = +\infty$ implies $x_k - x_0 > l$, $y_k = -\infty$ implies $x_k - x_0 < -l$ and $|y_k| < \infty$ implies $y_k = x_k$. It holds that $d_l^0(x) = d_l^0(y)$ and hence, $d(x, y) \leqslant 2^{-l}$ for an infinite number of configurations $x \in E$. □

**Corollary 7.** *Open balls are clopen (*i.e. *closed and open) sets.*

**Proof.** Any ball $[w]_r$ is, by definition, open. Let

$$W = \left\{ u \in \widetilde{[\![-r, r]\!]}^{[\![-r,r]\!]^d},\ u_0 = w_0, u \neq w \right\}$$

be the set of all cylinders of radius $r$ and of reference $w_0$ different from $w$. We have that $[w] = \mathfrak{C}_{w_0} \setminus \bigcup_{v\in W}[v]$. By Proposition 6, $\mathfrak{C}_{w_0}$ is compact and hence closed. We conclude that $[w]$ is a closed set minus some open sets, therefore it is closed. □

**Corollary 8.** *The space $\mathfrak{C}$ is locally compact (any open neighborhood of any point $x$ contains a compact neighborhood of $x$) and thus complete.*

**Proof.** Each point $x$ belongs to the open set $\mathfrak{C}_{x_0}$ which, by Proposition 6, is compact. Hence, for any open neighborhood $\mathcal{O}$ of $x$, $\mathcal{O} \cap \mathfrak{C}_{x_0}$ is open and contains an open ball $\mathcal{B}$ which contains $x$. Using Corollary 7, $\mathcal{B}$ is a closed subset of the compact $\mathfrak{C}_{x_0}$, and hence compact. □

**Corollary 9.** *The topological space $\mathfrak{C}$ is totally disconnected.*

**Proof.** Consider two distinct configurations $x$ and $y$. Let $d(x, y) = 2^{-r}$, and $B_x = [d_r^0(x)]_r$ be the open ball of center $x$ and radius $2^{-r}$. Since $B_x$ is clopen (Corollary 7), its complementary is open. Hence, $\mathfrak{C}$ is the union of $B_x$ which contains $x$ and its complementary (which contains $y$). Both sets are open. We conclude that $x$ and $y$ are in two distinct connected components and hence that $\mathfrak{C}$ is totally disconnected. □

Fig. 4. Basic behavior of $\mathcal{S}$.

## 3. Sand automata

A sand automaton (SA) is a deterministic finite automaton working on configurations. Each site is updated according to a local rule which computes the new sand content for the site taking into account its current sand content and the one of a fixed number of neighboring sites (the *range*). All sites are updated in parallel. The *radius* of the automaton is the maximal number of grains that it can add to or delete from a site.

**Definition 10** (*Range*). A *range* is a cylinder whose reference is unspecified. More formally for any configuration $x \in \mathfrak{C}$, $r \in \mathbb{N} \setminus \{0\}$ and $i \in \mathbb{Z}^d$ let $R_r^i(x)$ be the matrix of dimension $d$, for $k \in [\![-r, r]\!]^d$,

$$R_r^i(x)_k = \begin{cases} \perp & \text{if } k = 0, \\ \beta_r^{x_i}(x_{i+k}) & \text{otherwise.} \end{cases}$$

The set of all ranges of radius $r$, *i.e.* the set of all matrices $m$ of $\widetilde{[\![-r, r]\!]}^{[\![-r,r]\!]^d}$ such that $m_0 = \perp$ is denoted $\mathfrak{R}_r$.

It is now possible to give a formal definition of a SA.

**Definition 11** (*Sand Automaton*). A SA is a couple $\mathcal{A} \equiv \langle r, \lambda \rangle$, where $r$ is the *radius* and $\lambda : \mathfrak{R}_r \mapsto [\![-r, r]\!]$ the *local rule* of the automaton. By means of the local rule, one can define the *global rule* $f : \mathfrak{C} \mapsto \mathfrak{C}$ as follows

$$\forall x \in \mathfrak{C},\ \forall i \in \mathbb{Z}^d,\ \ f(x)_i = \begin{cases} x_i & \text{if } x_i = \pm\infty, \\ x_i + \lambda(R_r^i(x)) & \text{otherwise.} \end{cases}$$

When no misunderstanding is possible, we will make no distinction between the global rule $f$ and the automaton $\mathcal{A}$ itself. Moreover, for simple automata, we may omit the $\perp$ symbol in ranges.

**Example 12. The automaton $\mathcal{S}$.**
This automaton simulates SPM [3,4] in dimension 1: $\mathcal{S} = \langle 1, \lambda_{\mathcal{S}} \rangle$, where

$$\forall a, b \in \widetilde{[\![-1, 1]\!]}, \quad \lambda_{\mathcal{S}}(a, b) = \begin{cases} +1 & \text{if } a = +\infty \text{ and } b \neq -\infty, \\ -1 & \text{if } a \neq +\infty \text{ and } b = -\infty, \\ \ \ 0 & \text{otherwise.} \end{cases}$$

Remark on the basic grain movement of $\mathcal{S}$: a grain falls to the column on its right when the height difference is bigger than 2 (Fig. 4). Its long-term behavior is illustrated by an example in Fig. 5, with an initial configuration containing 7 grains in one single pile. After a finite number of iterations the system reaches a fixed point.

**Example 13. The automaton $\mathcal{S}^r$.**
This automaton is defined similarly to $\mathcal{S}$, but grains climb the cliffs instead of falling down. Can you simulate all sandpiles even on graphs (see Fig. 6)? Let $\mathcal{S}^r = \langle 1, \lambda_{\mathcal{S}^r} \rangle$ where

$$\forall a, b \in \widetilde{[\![-1, 1]\!]}, \quad \lambda_{\mathcal{S}^r}(a, b) = \begin{cases} -1 & \text{if } a = +\infty \text{ and } b \neq -\infty, \\ +1 & \text{if } a \neq +\infty \text{ and } b = -\infty, \\ \ \ 0 & \text{otherwise.} \end{cases}$$

**Remark 14.** The automaton $\mathcal{S}^r$ is the right inverse of $\mathcal{S}$.

Note that the model IPM introduced to study integer partitions [10] cannot be simulated as such because its sliding rule is not local. But the IPM($k$) extensions [13] can easily be simulated by a sand automaton of radius $k$.

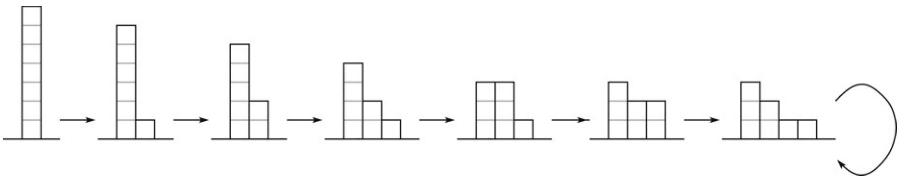


Fig. 5. Space–time diagram of $\mathcal{S}$, started from a single pile of height 7.

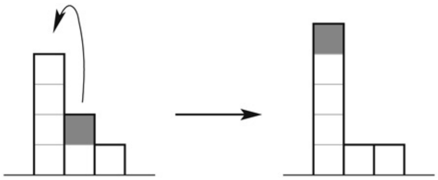


Fig. 6. Basic behavior of $\mathcal{S}^r$.

## 4. A Hedlund-like theorem

In this section, we prove a result which recalls Hedlund's theorem in cellular automata theory [16]. This allows us to link the computer science point of view, which is based on the finite description of the local rule of SA and on the notion of simulation, to the mathematics point of view which is essentially based on the global rule and on the notion of discrete dynamical system.

For all integers $k$ between 0 and $d-1$, let $\mathbb{1}_k$ be the vector all of whose coordinate are 0 except the $k$th which is 1. The $k$th *shift map* $\sigma_k : \mathfrak{C} \mapsto \mathfrak{C}$ is defined by $\forall x \in \mathfrak{C}, \forall i \in \mathbb{Z}^d, \sigma_k(x)_i = x_{i+\mathbb{1}_k}$. In dimension one, there is only one shift map noted $\sigma$. For any $n \in \mathbb{Z}^d$, let us define the function $\sigma^n : \mathfrak{C} \mapsto \mathfrak{C}$ by $\forall x \in \mathfrak{C}, \forall i \in \mathbb{Z}^d, \sigma^n(x) = x_{i+n}$. The *raising map* $\rho : \mathfrak{C} \mapsto \mathfrak{C}$ is defined by $\forall x \in \mathfrak{C}, \forall i \in \mathbb{Z}^d, \rho(x)_i = x_i + 1$.

A function $f : \mathfrak{C} \mapsto \mathfrak{C}$ is *shift-commuting* (resp. *vertical-commuting*) if $\forall k, f \circ \sigma_k = \sigma_k \circ f$ (resp. $f \circ \rho = \rho \circ f$).

**Definition 15** (*Infiniteness Conserving*). A function $f$ from $\mathfrak{C}$ to $\mathfrak{C}$ is *infiniteness conserving* if

$$\forall x \in \mathfrak{C},\ \forall i \in \mathbb{Z}^d, \quad \begin{cases} f(x)_i = +\infty \Leftrightarrow x_i = +\infty \\ \quad\quad \text{and} \\ f(x)_i = -\infty \Leftrightarrow x_i = -\infty. \end{cases}$$

**Lemma 16.** *Let $f : \mathfrak{C} \mapsto \mathfrak{C}$ be a continuous, vertical-commuting and infiniteness conserving function. Then, $\forall u \in \widetilde{\mathbb{Z}},\ f^{-1}(\mathfrak{C}_u)$ is compact.*

**Proof.** Let $f$ be a continuous, vertical-commuting and infiniteness conserving function.

By infiniteness conservation, if $u \in \widetilde{\mathbb{Z}}$ is infinite, we have $f^{-1}(\mathfrak{C}_u) \subseteq \mathfrak{C}_u$. By Proposition 6, $\mathfrak{C}_u$ is compact and, by continuity of $f$, $f^{-1}(\mathfrak{C}_u)$ is closed. We conclude that $f^{-1}(\mathfrak{C}_u)$ is compact since it is a closed subset of compact set.

Now, assume that $u$ is finite. Let $U = f^{-1}(\mathfrak{C}_u)$. Let $U_i = U \cap \mathfrak{C}_i$ and $I = \{i \in \mathbb{Z},\ U_i \neq \emptyset\}$.

We prove by contradiction that $I$ has finite cardinality. Assume $|I| = \infty$. For all $i$ in $I$, choose $x^i$ in $\rho^{-i}(U_i)$. Note that $x^i \in \mathfrak{C}_0$.

As $\mathfrak{C}_0$ is compact, the sequence $(x^i)_{i\in I}$ has a subsequence $(x^{\varsigma(n)})_{n\in\mathbb{N}}$ converging to $\ell$. As $f$ is continuous, $\lim_{n\to\infty} f(x^{\varsigma(n)}) = f(\ell)$ and so, if $v$ is such that $f(\ell) \in \mathfrak{C}_v$, there is an integer $N$ such that for all $n \geqslant N$, $f(x^{\varsigma(n)}) \in \mathfrak{C}_v$.

Let $n_1 = \varsigma(N)$ and $n_2 = \varsigma(N+1)$. Note that $\mathfrak{C}_x$ and $\mathfrak{C}_y$ are disjoint for $x \neq y$. As $f$ is vertical-commuting, $f(\rho^{n_1}(x^{n_1})) \in \mathfrak{C}_{v+n_1} \cap \mathfrak{C}_u$ and $f(\rho^{n_2}(x^{n_2})) \in \mathfrak{C}_{v+n_2} \cap \mathfrak{C}_u$. We deduce that $v + n_1 = u = v + n_2$ which is a contradiction since $\varsigma$ is injective.

Therefore, $I$ has finite cardinality. Since $U \subset \bigcup_{i\in I} \mathfrak{C}_i$ we have that $U$ is a closed set included in a finite union of compact sets. We conclude that $U$ is compact. $\square$

The following theorem is a strong representation result that characterizes a wide class of functions that have finite description on $\mathfrak{C}$. The advantage of such functions is that they are really suitable for computer simulations. The finite description allows a faultless computation of the values of the function reducing the sensibility to approximations errors which can completely bias simulations.

**Theorem 17.** *A function $f : \mathfrak{C} \mapsto \mathfrak{C}$ is the global function of a sand automaton if and only if the following conditions hold:*

(i) *$f$ is continuous;*
(ii) *$f$ is shift-commuting;*
(iii) *$f$ is vertical-commuting;*
(iv) *$f$ is infiniteness conserving.*

**Proof.** Consider a SA $\mathcal{A} = \langle r, \lambda \rangle$ with global rule $f$. It follows immediately by the definition of SA that $f$ is shift-commuting, vertical-commuting and infiniteness conserving. It remains to prove that $f$ is continuous. For any configuration $x$ and any positive integer $l$, we have to find an integer $m$ such that $d(x, y) < 2^{-m}$ implies $d(f(x), f(y)) < 2^{-l}$ for all configurations $y$. Choose $m = 3r + l$. Let $y$ be a configuration such that $d(x, y) < 2^{-m}$. We have $d_m^0(x) = d_m^0(y)$, which implies, on the one hand, $f(x)_0 = f(y)_0$, and on the other hand, $\beta_m^{x_0}(x_i) = \beta_m^{y_0}(y_i)$, for any vector $i \in [\![-m, m]\!]^d$.

We claim that for any vector $j \in [\![-l, l]\!]^d$, $j \neq 0$ we have two possible cases:

(i) $|x_j - x_0| > 2r + l$ and hence $|y_j - y_0| > 2r + l$;
(ii) $x_j = y_j$ and for all vector $k \in \{j + t,\ t \neq 0,\ |t| \leqslant r\}$, $\beta_r^{x_j}(x_k) = \beta_r^{y_j}(y_k)$.

Suppose that, on the one hand, $|x_j - x_0| > 2r + l$. Then, it holds that $|y_j - y_0| > 2r + l$, since $\beta_m^{x_0}(x_j) = \beta_m^{y_0}(y_j)$.

On the other hand, assume that $|x_j - x_0| \leqslant 2r + l$. We have that $x_j = y_j$, since $\beta_m^{x_0}(x_j) = \beta_m^{y_0}(y_j)$. Then, for all vectors $k \in \{j + t,\ t \neq 0,\ |t| \leqslant r\}$, we have three possible subcases (since $\beta_m^{x_0}(x_k) = \beta_m^{y_0}(y_k)$):

(a) $x_k = y_k$ then $\underline{\beta_r^{x_j}(x_k) = \beta_r^{y_j}(y_k)}$;
(b) $x_k - x_0 > m = 3r + l$ and $y_k - y_0 > m = 3r + l$. As $|x_j - x_0| \leqslant 2r + l$ and $|y_j - y_0| \leqslant 2r + l$ (recall that $x_j = y_j$ and $x_0 = y_0$), then $x_k - x_j > r$ and $y_k - y_j > r$ which implies that $\underline{\beta_r^{x_j}(x_k) = \beta_r^{y_j}(y_k)}$;
(c) else $x_k - x_0 < -m = -3r - l$ and $y_k - y_0 < -m = -3r - l$. Using the same chain of inequalities, it holds that $\underline{\beta_r^{x_j}(x_k) = \beta_r^{y_j}(y_k)}$

(in the formulas above and the text following the proof, we have underlined some parts to stress that they are equal).

We conclude that, for all integers $j \in [\![-l, l]\!]^d$ with $j \neq 0$, if case (i) occurs, since the local rule can increase or decrease a value by at most $r$, it holds that

$$|f(x)_j - f(x)_0| \geqslant |x_j - x_0| - |f(x)_0 - x_0| - |f(x)_j - x_j| > (2r + l) - r - r = l$$

and, by using the same argument, one finds $|f(y)_j - f(y)_0| > l$. Therefore, it holds that $\underline{\beta_l^{f(x)_0}(f(x)_j) = \beta_l^{f(y)_0}(f(y)_j)}$.

If case (ii) occurs, then $x_j = y_j$ and $d_r^j(x) = d_r^j(y)$. That means that $f(x)_j = f(y)_j$, and so $\underline{\beta_l^{f(x)_0}(f(x)_j) = \beta_l^{f(y)_0}(f(y)_j)}$.

Hence, it holds that $d_l^0(f(x)) = d_l^0(f(y))$, and then $d(f(x), f(y)) < 2^{-l}$.

For the second part of the proof, let $f : \mathfrak{C} \mapsto \mathfrak{C}$ be a continuous, shift-commuting, vertical-commuting and infiniteness conserving function. We are going to prove that it is the global rule of a suitable SA.

Consider the clopen set $\mathfrak{C}_0$. Let $U = f^{-1}(\mathfrak{C}_0)$. By Lemma 16, the set $U$ is compact, and, hence, it is a union of finitely many open balls: $U = \bigcup_{i \in I} [w^i]_{r_i}$ with $|I| < \infty$. Since each ball can be decomposed into finitely many balls of larger radius, without loss of generality, one can suppose that each cylinder $w^i$ has the same radius $r$.

In the following, the range obtained from a cylinder $w$ whose reference value has been erased is denoted $\langle\!\langle w \rangle\!\rangle$. Suppose that for $i \neq j$, $\langle\!\langle w^i \rangle\!\rangle = \langle\!\langle w^j \rangle\!\rangle$. Then, let $a = w_0^i$ and $b = w_0^j$. As the cylinders $w^i$ and $w^j$ are distinct but have the same range, $a \neq b$. Choose $x$ in $[w^i]$, and let $y = \rho^{b-a}(x)$. We have $y \in [w^j]$. Using vertical invariance of $f$, one has $f(y) \in \mathfrak{C}_0 \cap \mathfrak{C}_{b-a} = \emptyset$, which is a contradiction.

Thus we have that every range must appear exactly once in the sequence $(\langle\!\langle w^i \rangle\!\rangle)_{i \in I}$.

Suppose now that a range $R$ does not appear in the sequence $(\langle\!\langle w^i \rangle\!\rangle)_{i\in I}$, let $x$ be a configuration such that $x_0 = 0$ and $R_r^0(x) = R$. The configuration $f(x)$ belongs to an $\mathfrak{C}_j$, for some finite $j$ since $f$ is infiniteness conserving. Hence, by vertical invariance, $f(\rho^{-j}(x)) \in \mathfrak{C}_0$. Since $R_r^0(\rho^{-j}(x)) = R_r^0(x) = R$, it means that $R$ appears in $\langle\!\langle w^i \rangle\!\rangle_{i\in I}$, which is a contradiction.

Hence, it is natural to define $\lambda$ as follows: $\lambda(\langle\!\langle w^i \rangle\!\rangle) = -w_0^i$. Let $f'$ be the global rule of the SA $\langle r, \lambda\rangle$. Let us prove that $f = f'$. For all configurations $x$ and for all vectors $n$, let $i \in I$ be such that $\langle\!\langle w^i \rangle\!\rangle = R_r^n(x)$. We have that $f'(x)_n = x_n - w_0^i$.

Let us compute $f(x)_n$. Since $f$ is vertical-commuting and shift-commuting, we have that

$$f(x)_n = f(\sigma^n(x))_0 = x_n - w_0^i + f(\rho^{w_0^i - x_n}(\sigma^n(x)))_0.$$

Let $y = \rho^{w_0^i - x_n}(\sigma^n(x))$. We have $d_r^0(y) = w^i$. Hence, by definition of $w^i$, $f(y)_0 = 0$. Hence, $f(x)_n = x_n - w_0^i + f(y)_0 = f'(x)$. We conclude that $f = f'$. $\square$

**Remark 18.** The last condition of this theorem is very important. It distinguishes SA from CA, ensuring that no "holes" can be created in a configuration.

The representation theorem allows us to prove a very interesting result, namely that the inverse of a SA is still a SA. The proof of this result needs the following necessary condition for injective SA.

**Proposition 19.** *Consider a* SA *of global rule $f$. If $f$ is injective then $f$ is open.*

**Proof.** Consider an injective SA $\langle r, \lambda\rangle$ of global rule $f$. Let $A$ be an open ball $[w]_l$, with $l > r$ (otherwise it can be seen as the union of a finite number of open balls of radius lower than $2^{-r}$). By definition of SA, $f(A) \subset \mathfrak{C}_{w_0+i}$ where $i$ is the result of the application of the $\lambda$ to $\langle\!\langle w \rangle\!\rangle$. Let $C = f^{-1}(\mathfrak{C}_{w_0+i})$ and $B = C\backslash A$. Since $f$ is injective, we have that $f(C) = \mathfrak{C}_{w_0+i}$ is the disjoint union of $f(A)$ and $f(B)$. Since $f$ is continuous, $C$ is clopen. As $A$ is also clopen, $B$ is clopen. Using Lemma 16, $C$ is included in a finite union of sets $\mathfrak{C}_i$ hence, by Proposition 6, it is compact. We deduce that, since $B$ is closed, it is compact, and, by the continuity of $f$, $f(B)$ is compact too. We conclude that $f(B)$ is closed, and hence, that $f(A) = f(C) \setminus f(B)$ is open since $f(C) = \mathfrak{C}_{w_0+i}$ is open. $\square$

**Proposition 20.** *Consider a* SA $\mathcal{A}$ *of global rule $f$. If $f$ is bijective, then $f^{-1}$ is a* SA.

**Proof.** Consider an injective SA $\mathcal{A}$ of global rule $f$. By Proposition 19, $f$ is open and, hence, $f^{-1}$ is continuous. Since $f$ is vertical-commuting so is $f^{-1}$: for all configurations $x$ and $y = f^{-1}(x)$, $f(\rho(x)) = \rho(f(x)) \Rightarrow f(\rho(f^{-1}(y)) = \rho(y) \Rightarrow \rho(f^{-1}(y)) = f^{-1}(\rho(y))$. Replacing $\rho$ by $\sigma$ gives that $f^{-1}$ is shift-commuting too. It is clear that $f^{-1}$ is infiniteness conserving. Hence, by Theorem 17, $f^{-1}$ is the global rule of a SA. $\square$

## 5. Relation to cellular automata

Cellular automata are often used as a paradigmatic example for modeling phenomena ruled by local interaction rules. Proposition 21 says that SA can be used as well. Cellular automata can be formally defined as follows.

For any finite set $S$, a *cellular automaton* is a map $F : S^{\mathbb{Z}^d} \mapsto S^{\mathbb{Z}^d}$ defined for any $x \in S^{\mathbb{Z}^d}$ and $i \in \mathbb{Z}^d$ by $F(x)_i = \mu(\mathbf{M}^i)$, where $\mathbf{M}^i \in S^{[\![-h,h]\!]^d}$ is the *neighborhood* of $x_i$, *i.e.* $\mathbf{M}^i_j = x_{i+j}$ for all $j \in [\![-h, h]\!]^d$. Moreover, $h$ is the *radius* and $\mu : S^{(2h+1)^d} \mapsto S$ is the *local rule*. The set $S$ is usually called the set of *states* of the cellular automaton. The function $F$ is called the *global rule* of the cellular automaton (for more on cellular automata, see [17] for example).

**Proposition 21.** *Any cellular automaton can be simulated by a suitable* SA.

**Proof.** We are going to consider only cellular automata with two states since any cellular automaton can be simulated by a suitable cellular automaton with state set $\{0, 1\}$. Moreover, we prove it for one-dimensional cellular automata only, higher dimensions are similar.

Consider a cellular automaton $\mathcal{C}$ in dimension 1, of local rule $\mu$, radius $r$, state set $S = \{0, 1\}$ and global rule $F$. A configuration $x \in \{0, 1\}^{\mathbb{Z}}$ will be coded by $\zeta(x) \in \mathfrak{C}$ as follows (see Fig. 7)

$$\forall n \in \mathbb{Z}, \zeta(x)_n = \begin{cases} x_{n/2} & \text{if } n \text{ is even,} \\ 2 & \text{otherwise.} \end{cases}$$

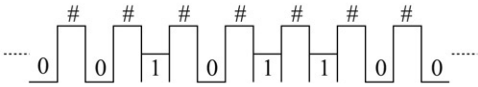



Fig. 7. An example of simulation of a cellular automaton by a SA, starting from the configuration . . . 00101100 . . . . The symbol # is a marker used by the SA for distinguishing the cellular automaton states.

Clearly, $\zeta$ is bijective and thus any configuration $x$ can be uniquely reconstructed from $\zeta(x)$. We will simulate $\mathcal{C}$ by the SA $\mathcal{A} = \langle 2r, \lambda\rangle$, where $\lambda$ is defined as follows

$$\lambda(w) = \begin{cases} \mu(w_{-2r}, w_{-2r+2}, \ldots, w_{-2}, 0, w_2, \ldots, w_{2r-2}, w_{2r}) \\ \qquad\qquad\qquad\qquad \text{if } w_{2i+1} = 2 \text{ for } -r < i < r-1 \\ \mu(w_{-2r}+1, w_{-2r+2}+1, \ldots, w_{-2}+1, 1, w_2+1, \ldots, \\ \qquad w_{2r-2}+1, w_{2r}+1) - 1 \\ \qquad\qquad\qquad\qquad \text{if } w_{2i+1} = 1 \text{ for } -r < i < r-1 \\ 0 \qquad\qquad\qquad\qquad \text{otherwise.} \end{cases}$$

Let $f$ be the global rule of $\mathcal{A}$. Using the third line in the definition of $\lambda$, one finds that

$$\forall n \in \mathbb{Z}, \quad f(\zeta(x))_{2n+1} = 2. \tag{1}$$

Moreover, $\forall n \in \mathbb{Z}$ and for all integers $i$ between $-2r$ and $2r$, let $w_i = \zeta(x)_{2n+i} - \zeta(x)_{2n}$. If $\zeta(x)_{2n} = 1$, then, using Eq. (1), $\forall i,\ -r < i < r-1$, $w_{2i+1} = 2 - 1 = 1$ and hence,

$$\begin{aligned} f(\zeta(x))_{2n} &= x_{2n} + \mu(w_{-2r}+1, w_{-2r+2}+1, \ldots, w_{2r-2}+1, w_{2r}+1) - 1 \\ &= \mu(\zeta(x)_{2n-2r}, \zeta(x)_{2n-2r+2}, \ldots, \zeta(x)_{2n}, \ldots \zeta(x)_{2n+2r}) \\ &= \mu(x_{n-r}, \ldots, x_{n+r}). \end{aligned}$$

If $\zeta(x)_{2n} = 0$, then, using Eq. (1), $w_{2i+1} = 2$ for $-r < i < r-1$, and hence,

$$\begin{aligned} f(\zeta(x))_{2n} &= x_{2n} + \mu(w_{-2r}, w_{-2r+2}, \ldots, w_{-2}, 0, w_2, \ldots, w_{2r-2}, w_{2r}) \\ &= \mu(\zeta(x)_{2n-2r}, \zeta(x)_{2n-2r+2}, \ldots, \zeta(x)_{2n}, \ldots \zeta(x)_{2n+2r}) \\ &= \mu(x_{n-r}, \ldots, x_{n+r}). \end{aligned}$$

We conclude that $f(\zeta(x)) = \zeta(F(x))$. □

**Proposition 22.** *Any* SA *can be simulated by a suitable cellular automaton.*

**Proof.** We give the proof for any one-dimensional SA, larger dimensions are similar. Such a SA is simulated by a two dimensional CA. A configuration $x \in \mathfrak{C}$ is coded by $\zeta(x) \in \{0,1\}^{\mathbb{Z}^2}$ as follows, as shown in Fig. 8:

$$\forall i, j \in \mathbb{Z}, \quad \zeta(x)_{i,j} = \begin{cases} 1 & \text{if } x_i > j, \\ 0 & \text{otherwise.} \end{cases}$$

More formally, the CA is defined as follows. Let $\mathcal{A} = \langle r, \lambda\rangle$ be a SA. The two-dimensional CA simulating $\mathcal{A}$ has radius $2r$, state set $\{0, 1\}$, and its local rule $\mu$ is defined as follows. Let $w$ be a one-dimensional range, which is a sequence $(w_{-r}, \ldots, w_{-1}, w_1, \ldots, w_r)$. Define $w_0 = 0$ for convenience. Let $n = \lambda(w)$, $-r \leqslant n \leqslant r$.

- If $n > 0$, for all $k \in [\![0, n-1]\!]$, for all CA neighborhoods $N$ of radius $r+n$ such that

  $$\forall i \in [\![-r, r]\!],\ \forall j \in [\![-k-r-1, -k+r]\!], \quad N_{i,j} = \begin{cases} 0 & \text{if } j \geqslant w_i - k, \\ 1 & \text{otherwise,} \end{cases}$$

  we set $\mu(N) = 1$.
- If $n < 0$, for all $k \in [\![n, -1]\!]$, for all neighborhoods $N$ of radius $r + |n|$ such that

  $$\forall i \in [\![-r, r]\!],\ \forall j \in [\![-k-r-1, -k+r]\!], \quad N_{i,j} = \begin{cases} 0 & \text{if } j \geqslant w_i - k, \\ 1 & \text{otherwise,} \end{cases}$$

  we set $\mu(N) = 0$.

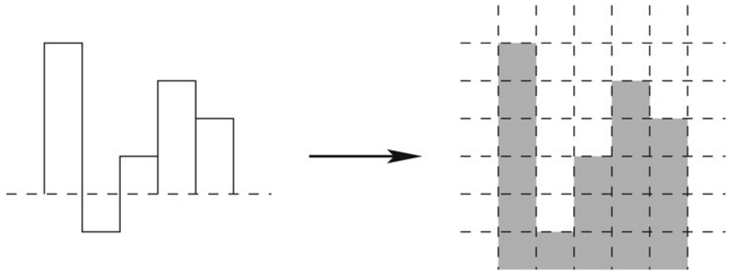


Fig. 8. Simulation of a SA by a cellular automaton. Example using the configuration (... 4 −1 1 3 2 ...) in dimension 1.

- For all other local neighborhoods of the CA, $\mu$ does not modify the state of the central cell. □

The following example illustrates the simulation described in the proof of Proposition 22.

**Example 23.** The sand automaton $\mathcal{S}$ defined in Section 3 can be easily simulated by a 2 states cellular automaton $\mathcal{C} = \langle 2, \mu \rangle$ in dimension 2. The case where $\lambda_{\mathcal{S}}$ returns $-1$ is expressed by the rules

$$\mu \begin{pmatrix} 0 & 0 & 0 \\ a & 0 & 0 \\ b & \mathbf{1} & 0 \\ c & 1 & 0 \\ d & 1 & e \end{pmatrix} = 0,$$

where $0 \leqslant a \leqslant b \leqslant c \leqslant d \leqslant 1$ and $0 \leqslant e \leqslant 1$. In the same way, $\lambda_{\mathcal{S}}$ returns 1 is expressed by

$$\mu \begin{pmatrix} a & 0 & b \\ 1 & 0 & c \\ 1 & \mathbf{0} & d \\ 1 & 1 & e \\ 1 & 1 & 1 \end{pmatrix} = 1,$$

where $0 \leqslant a \leqslant 1$ and $0 \leqslant b \leqslant c \leqslant d \leqslant e \leqslant 1$. For all other inputs, $\mu$ does not change the central value.

**Remark 24.** Our simulation of a CA by a SA doubles the initial radius. It could also be done by increasing the radius by one (only one marker is needed in the neighborhood), but the simulation would be slightly more complex.

For the simulation of a SA by a CA, the radius is doubled and the dimension increased by 1. The increase of the dimension is necessary to code the unbounded number of states and still be able to simulate the iterations in constant time. Then, once a dimension is added, the radius has to be doubled as was done in the proof of Proposition 22.

## 6. Basic set properties of sand automata

In this section we begin the study of our model, in the same way as it was done in [15] for cellular automata. We study the relations between surjectivity and injectivity, *w.r.t.* all finite and periodic configurations (see [2]). This leads to some results which help understanding the basic behavior of this model, before looking for more complex dynamic properties.

A configuration $x$ is *finite* if $\exists k \in \mathbb{N}$ such that for any vector $i \in \mathbb{Z}^d$, $|i| \geqslant k \Rightarrow x_i = 0$ and $|i| < k \Rightarrow |x_i| < \infty$. The set of finite configurations is noted $\mathfrak{F}$. For any finite configuration $x$, the *size* of $x$ is $|x| = \max_{i,j \in \mathbb{Z}^d}\{|i-j|,\ x_i \neq 0 \text{ and } x_j \neq 0\}$. A configuration $x$ is (spatially) *periodic* if there is a vector $p = (p_1, \ldots, p_d) \in \mathbb{N}^d$ (called the *period*) such that for any vector $i = (i_1, \ldots, i_d) \in \mathbb{Z}^d$, for any integers $t_1, \ldots, t_d \in \mathbb{Z}$, $x_i = x_{i_1+t_1p_1,\ldots,i_d+t_dp_d}$ and $|x_i| < \infty$. $\mathfrak{P}$ denotes the set of (spatially) periodic configurations.

The SA $\mathcal{A}$ is *surjective* [resp. *injective*] if its global rule $f$ is surjective [resp. injective]. For any set $\mathfrak{U} \subseteq \mathfrak{C}$, $f$ is said to be $\mathfrak{U}$-surjective [resp. injective] if the restriction of $f$ to $\mathfrak{U}$ is surjective [resp. injective].

In the definitions of finite and periodic configurations, we arbitrarily decided to remove sources and sinks. Consider $\widetilde{\mathfrak{F}}$ defined by $x \in \widetilde{\mathfrak{F}}$ iff there exists $k \in \mathbb{N}$ such that for all $i \in \mathbb{Z}^d$, $|i| \geqslant k \Rightarrow x_i = 0$; and $\widetilde{\mathfrak{P}}$ defined by $x \in \widetilde{\mathfrak{P}}$ iff there is a vector $p \in \mathbb{N}^d$ such that for all $i, t \in \mathbb{Z}^d$, $x_i = x_{i_1+t_1p_1,\dots,i_d+t_dp_d}$ (*i.e.* we allow $|x_i| = \infty$ in the finite or periodic configuration). The following proposition allows us to study basic set properties only on the sets $\mathfrak{F}$ and $\mathfrak{P}$.

**Proposition 25.** *A* SA *is $\mathfrak{F}$-surjective [resp. $\mathfrak{F}$-injective] iff it is $\widetilde{\mathfrak{F}}$-surjective [resp. $\widetilde{\mathfrak{F}}$-injective], and $\mathfrak{P}$-surjective [resp. $\mathfrak{P}$-injective] iff it is $\widetilde{\mathfrak{P}}$-surjective [resp. $\widetilde{\mathfrak{P}}$-injective].*

**Proof.** We prove these equivalences over $\mathfrak{F}$ and $\widetilde{\mathfrak{F}}$, similar proofs can be done for the periodic configurations over $\mathfrak{P}$ and $\widetilde{\mathfrak{P}}$. First, we show that $\widetilde{\mathfrak{F}}$-surjectivity implies $\mathfrak{F}$-surjectivity. Let $f$ be the global rule of a $\widetilde{\mathfrak{F}}$-surjective SA, and let $x \in \mathfrak{F}$. Then $x \in \widetilde{\mathfrak{F}}$ also, and there is $y \in \widetilde{\mathfrak{F}}$ such that $f(y) = x$. As there are no infinite columns in $x$, and $f$ is infiniteness conserving, there are no infinite columns in $y$ so $y \in \mathfrak{F}$.

Conversely, let $f$ be the global rule of a $\mathfrak{F}$-surjective SA, let $x \in \widetilde{\mathfrak{F}}$. If $x \in \mathfrak{F}$, its pre-image is in $\mathfrak{F} \subset \widetilde{\mathfrak{F}}$. Otherwise, let $x' \in \mathfrak{F}$ defined by

$$\forall i \in \mathbb{Z}^d, \quad x'_i = \begin{cases} x_i & \text{if } |x_i| < \infty, \\ M + 3r + 1 & \text{if } x_i = +\infty, \\ m - 3r - 1 & \text{if } x_i = -\infty, \end{cases}$$

where $M = \max_{i\in\mathbb{Z}^d}\{x_i,\ |x_i| < \infty\}$ and $m = \min_{i\in\mathbb{Z}^d}\{x_i,\ |x_i| < \infty\}$. Let $y' \in \mathfrak{F}$ such that $f(y') = x'$, and $y \in \widetilde{\mathfrak{F}}$ defined by

$$\forall i \in \mathbb{Z}^d, \quad y_i = \begin{cases} y'_i & \text{if } |x_i| < \infty, \\ +\infty & \text{if } x_i = +\infty, \\ -\infty & \text{if } x_i = -\infty. \end{cases}$$

Then, it holds that

$$\forall i \in \mathbb{Z}^d, \quad f(y)_i = \begin{cases} f(y')_i = x'_i = x_i & \text{if } |x_i| < \infty, \\ +\infty & \text{if } x_i = +\infty, \\ -\infty & \text{if } x_i = -\infty. \end{cases}$$

Only the equality $f(y)_i = f(y')_i$ is not obvious. It is justified by the fact that in $y$ and $y'$ the same neighborhoods are seen. Indeed, if there is an infinite column in the neighborhood of $y$, it is of height at least $M + 2r + 1$ or at most $m - 2r - 1$ in $y'$. This is in any way out of sight of the measuring device of reference $y'_i$, whose height is in the interval $[\![-m - r, M + r]\!]$. Therefore the height of the column $i$ is considered infinite and hence, $f$ is $\widetilde{\mathfrak{F}}$-surjective.

Suppose that $f$ is the global rule of a $\widetilde{\mathfrak{F}}$-injective automaton. Let $x^1, x^2 \in \mathfrak{F} \subset \widetilde{\mathfrak{F}}$, $x^1 \neq x^2$, it holds that $f(x^1) \neq f(x^2)$ as $f$ is $\widetilde{\mathfrak{F}}$-injective, hence it is also $\mathfrak{F}$-injective.

Conversely, let $f$ be the global rule of a $\mathfrak{F}$-injective SA, and let $x^1, x^2 \in \widetilde{\mathfrak{F}}$, $x^1 \neq x^2$. As we did previously, we replace infinite columns by $M + r + 1$ and $m - r - 1$ in $x^1$ and $x^2$ to build $x^{1'}, x^{2'} \in \mathfrak{F}$ ($M = \max(M^1, M^2)$ and $m = \min(m^1, m^2)$). There is an index $i \in \mathbb{Z}^d$ such that $f(x^{1'})_i \neq f(x^{2'})_i$ because $f$ is $\mathfrak{F}$-injective. Then we have the following cases:

(i) if $|x^1_i| < \infty$ and $|x^2_i| < \infty$, it holds that $f(x^1)_i = f(x^{1'})_i \neq f(x^{2'})_i = f(x^2)_i$ ;
(ii) if $x^1_i = +\infty$, necessarily $x^2_i \neq +\infty$ otherwise $x^{1'}_i = x^{2'}_i = M + r + 1$ and $f(x^{1'})_i = f(x^{2'})_i$ (the neighborhood is $-\infty$ everywhere in both configurations, the equality is preserved), hence, $f(x^1)_i = +\infty \neq f(x^2)_i$ ;
(iii) if $x^1_i = -\infty$ or $x^2_i = \pm\infty$, for the same reasons $f(x^1)_i \neq f(x^2)_i$.

In all cases, $f(x^1) \neq f(x^2)$, therefore $f$ is $\widetilde{\mathfrak{F}}$-injective too. $\square$

**Proposition 26.** *$\mathfrak{P}$-surjectivity implies surjectivity.*

**Proof.** For any configuration $x$, let $x^n \in \widetilde{\mathfrak{P}}$ be the $(2n+1, \dots, 2n+1)$-periodic configuration such that $\forall i \in \mathbb{Z}^d$, $|i| \leqslant n$, $(x^n)_i = x_i$. Consider a sand automaton $f$ that is $\mathfrak{P}$-surjective and choose an arbitrary configuration $x \in \mathfrak{C}$. For any $n \in \mathbb{N}$, let $y^n = f^{-1}(x^n) \in \widetilde{\mathfrak{P}}$ (from Proposition 25, $f$ is also $\widetilde{\mathfrak{P}}$-surjective). The pre-images $y^n$ are contained in some set $\mathfrak{C}_u$ for $u \in U = [\![x_0 - r, x_0 + r]\!]$ where $r$ is the precision of $f$. Since $\cup_{u\in U}\mathfrak{C}_u$ is compact and $(y^n)_{n\in\mathbb{N}} \subset \cup_{u\in U}\mathfrak{C}_u$, $(y^n)_{n\in\mathbb{N}}$ contains a converging sub-sequence $(y^{n_k})_{k\in\mathbb{N}}$. Let $y = \lim_{k\to\infty} y^{n_k}$. By contradiction, assume that $f(y) \neq x$. Then there exists $j \in \mathbb{Z}$ such that $f(y)_j \neq x_j$ but $f(y^{n_k})_j = x_j$ for $n_k$ big enough. $\square$

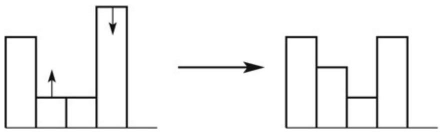


Fig. 9. Basic behavior of $\mathcal{L}$.

**Proposition 27.** *$\mathfrak{F}$-surjectivity implies surjectivity.*

**Proof.** This is roughly the same proof as for Proposition 26. The only change is that it starts with $x^n \in \widetilde{\mathfrak{F}}$ defined as the finite configuration with $\forall i \in \mathbb{Z}^d$, $(x^n)_i = x_i$ if $|i| \leqslant n$, and $(x^n)_i = 0$ otherwise. Everything else is unchanged. $\square$

The following result shows that the converse of Proposition 27 is false.

**Proposition 28.** *There is a sand automaton which is $\mathfrak{P}$-surjective (hence surjective by Proposition 26) but not $\mathfrak{F}$-surjective.*

**Proof.** Consider the automaton $\mathcal{L} = \langle 1, \lambda_{\mathcal{L}} \rangle$ where

$$\forall a, b \in \widetilde{[\![-1, 1]\!]}, \quad \lambda_{\mathcal{L}}(a, b) = \begin{cases} -1 & \text{if } a < 0, \\ +1 & \text{if } a > 0, \\ \ \ 0 & \text{otherwise.} \end{cases}$$

Fig. 9 illustrates the basic behavior of $\mathcal{L}$: each column tries to reach the height of its left neighbor.

Let us prove that $\mathcal{L}$ is not $\mathfrak{F}$-surjective. Consider the finite configuration $x$ where $x_0 = 2$ and $x_i = 0$ if $i \neq 0$. By contradiction, assume that $y$ is the pre-image of $x$ and that $y \in \mathfrak{F}$. Let $i$ be the greatest integer such that $y_i \neq 0$. Then since $y_i \neq 0$ and $y_{i+1} = 0$, it holds that $f_{\mathcal{L}}(y)_{i+1} = x_{i+1} \neq 0$. This implies that $i = -1$ because $x_0$ is the only non-zero value in $x$. But in that case, we have $y_0 = 0$, and as $\lambda_{\mathcal{L}}$ cannot return more than 1, $x_0 = 2$ cannot be reached. This is a contradiction.

To complete the proof, let us show that $\mathcal{L}$ is $\mathfrak{P}$-surjective. Choose an arbitrary configuration $x \in \mathfrak{P}$ of period $p \in \mathbb{N}$, we are going to build one of its periodic pre-images $y$. There is a unique sequence of strictly increasing indices $(i_n)_{n \in [\![0,k]\!]}$, $k < p$, such that $\forall i \in [\![i_n, i_{n+1}[\![$, $x_i = x_{i_n}$ and $x_{i_n} \neq x_{i_n - 1}$ (every $i_n$ corresponds to a variation of height in $x$). The idea is to work on these intervals, amplifying the difference at the border so that an application of the rule corrects it. Formally, if $k < 0$, nothing is done, $x$ is its own periodic pre-image. Otherwise for every $i \in [\![i_0, p + i_0 - 1]\!]$, let $n \in [\![0, k]\!]$ be such that $i_n \leqslant i < i_{n+1}$ (define $i_{k+1} = i_0 + p$), and assume that $x_{i_n - 1} < x_{i_n}$ (if it is not the case then the symmetrical operations have to be performed). Let $y_i = x_i + 1$ if $i - i_n$ is even, $y_i = x_i - 1$ if $i - i_n$ is odd. This construction has to be repeated on the other periods of $x$, giving the same results so $y$ is also $p$-periodic.

Clearly $f_{\mathcal{L}}(y) = x$. Indeed, for every $i \in [\![i_0, p + i_0 - 1]\!]$, first suppose that there is a $n \in [\![0, k]\!]$ such that $i = i_n$. We have $f_{\mathcal{L}}(y)_i = y_i + \lambda_{\mathcal{L}}(R_1^i(y))$. Supposing that $x_{i-1} < x_i$ (again, if it is the opposite then the operations are symmetrical), we have $y_i = x_i + 1 > x_{i-1} + 1$, hence, $y_i > y_{i-1}$ since $|x_{i-1} - y_{i-1}| \leqslant 1$ . So $\lambda_{\mathcal{L}}(R_1^i(y)) = -1$, and $f_{\mathcal{L}}(y)_i = x_i + 1 - 1 = x_i$. Otherwise if $i \neq i_n$ for all $n \in [\![0, k]\!]$, then by construction we have either:

(i) $y_i = x_i + 1$ and $y_{i-1} = x_{i-1} - 1 = x_i - 1$, because $x$ is constant between the $i_n$'s. Hence, $y_{i-1} = y_i - 2$, and then $f_{\mathcal{L}}(y)_i = x_i + 1 - 1 = x_i$;

(ii) or $y_i = x_i - 1$ and $y_{i-1} = x_{i-1} + 1$, the same method gives the result.

The configurations $x$ and $f_{\mathcal{L}}(y)$ are $p$-periodic, hence, they are also equal outside this interval. We conclude that $\mathcal{L}$ is $\mathfrak{P}$-surjective. $\square$

**Proposition 29.** *$\mathfrak{P}$-injectivity implies $\mathfrak{F}$-injectivity.*

**Proof.** This is proved using the contrapositive. Let $\mathcal{A}$ be an automaton not $\mathfrak{F}$-injective. Let $x^1, x^2$ be the two distinct finite configurations which lead to the same image $z$. Let $k \in \mathbb{N}$ such that for all $i \in \mathbb{Z}^d$, $|i| > k$, $x_i^1 = x_i^2 = 0$. We are going to build two distinct periodic configurations by surrounding the non-zero part of $x^1$ and $x^2$ with a crown of zeros, of thickness $r$, and repeating this pattern (see Fig. 10 for an illustration in dimension 2).

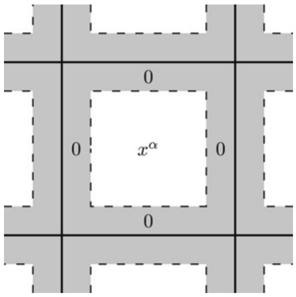



Fig. 10. Construction of $y^\alpha$ in dimension 2. White is for non-zero values taken in $x^\alpha$, gray is for 0.

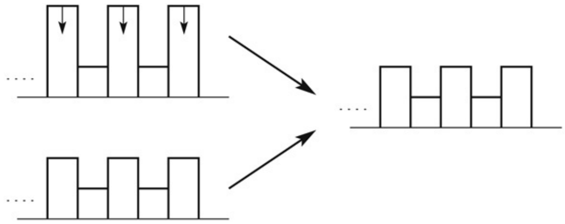


Fig. 11. Examples of evolution of $\mathcal{X}$ on two different configurations.

For $\alpha \in \{1, 2\}$, let $y^\alpha$ be the $(2k + 2r + 1, \ldots, 2k + 2r + 1)$-periodic configuration defined by

$$\forall i \in \mathbb{Z}^d, |i| \leqslant k + r, \qquad \begin{cases} y_i^\alpha = x_i^\alpha & \text{if } |i| \leqslant k, \\ y_i^\alpha = 0 & \text{if } k < |i| \leqslant k + r. \end{cases}$$

We have $f(y^1) = f(y^2)$. For every configuration, we can consider the translated configuration whose index is lower in norm than $k + r$ because of the periodicity. This configuration reacts as it did in $x^1$ and $x^2$ because its neighborhood is the same : inside the $k$ "circle", it is obvious. If it is inside the crown of 0's, then the only non-zero values it can see are the values located inside the initial pattern. So its behavior is equivalent to the one of the point at the border of the initial finite configuration, and $\mathcal{A}$ is not $\mathfrak{P}$-injective. □

The converse of Proposition 29 is false, as shown by the following proposition.

**Proposition 30.** *There is a sand automaton which is $\mathfrak{F}$-injective but neither injective nor $\mathfrak{P}$-injective.*

**Proof.** Consider the sand automaton $\mathcal{X} = \langle 2, \lambda_{\mathcal{X}} \rangle$ where

$$\forall a, b, c \in \widetilde{[\![-2, 2]\!]}, \qquad \begin{aligned} &\lambda_{\mathcal{X}}(+\infty, a, b, c) = -1, \\ &\lambda_{\mathcal{X}}(2, a, b, c) = -1, \\ &\lambda_{\mathcal{X}}(1, -1, a, b) = -1, \\ &\lambda_{\mathcal{X}}(1, -2, a, b) = -1, \\ &\lambda_{\mathcal{X}}(1, -\infty, a, b) = -1, \\ &\lambda_{\mathcal{X}}(0, -2, a, b) = -1, \\ &\lambda_{\mathcal{X}}(0, -\infty, a, b) = -1, \end{aligned}$$

and any other value gives 0. The behavior of this automaton on two specific sequences used in this proof is shown in Fig. 11. The evolutions of $\mathcal{X}$ on more general configurations seem quite hard to describe. Anyway, in the following we will need to study its evolutions only on special (simple) configurations.

Let us show that $\mathcal{X}$ is $\mathfrak{F}$-injective but neither injective nor $\mathfrak{P}$-injective. Consider the two periodic configurations $x$ and $y$ defined as follows (see Fig. 11):

$$\forall i \in \mathbb{Z}, \qquad \begin{cases} x_{2i} = 0 \\ x_{2i+1} = 1, \end{cases} \qquad \begin{cases} y_{2i} = 0 \\ y_{2i+1} = 2. \end{cases}$$

It can be easily verified that $f_{\mathcal{X}}(x) = f_{\mathcal{X}}(y) = x$. Hence, $\mathcal{X}$ is not $\mathfrak{P}$-injective and, of course, it is not injective.

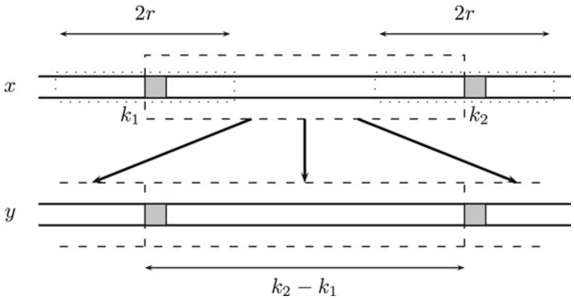



Fig. 12. Construction of $y$ using $x$.

Let us prove that $\mathcal{X}$ is $\mathfrak{F}$-injective. Let $x$ and $y$ be two distinct finite configurations, and suppose that their image by $f_{\mathcal{X}}$ is identical. As the two configurations are finite, we can define $i \in \mathbb{Z}$ being the least integer such that $x_i \neq y_i$. As $\lambda_{\mathcal{X}}$ returns only 0 or $-1$, we know that $|x_i - y_i| = 1$, and we can suppose that $x_i = y_i + 1$. That means that the local rule applied to $x$ at position $i$ is one of the seven cases which return $-1$:

- if the neighborhood is $(+\infty, -, -, -)$ (to make the notations clearer, $-$ represents any value), then since $y_i = x_i - 1$ and $y_{i-2} = x_{i-2}$, the same rule is applied to $y$, which means that $f_{\mathcal{X}}(x)_i \neq f_{\mathcal{X}}(y)_i$ which is a contradiction;
- if the neighborhood is $(2, -, -, -)$, for the same reason the rule for the neighborhood $(+\infty, -, -, -)$ is applied to $y$, which raises the same contradiction;
- again, if the neighborhood is $(1, -1, -, -)$, $(1, -2, -, -)$ or $(1, -\infty, -, -)$, the rule for the neighborhood $(2, -, -, -)$ is applied to $y$, making $y_i$ decrease by 1: same contradiction;
- if the neighborhood is $(0, -2, -, -)$ or $(0, -\infty, -, -)$, because $y_{i-2} = x_{i-2}$, $y_{i-1} = x_{i-1}$ and $y_i = x_i - 1$, one of the rules corresponding to the neighborhoods $(1, -1, -, -)$, $(1, -2, -, -)$ or $(1, -\infty, -, -)$ is applied to $y$. Again, we have $f_{\mathcal{X}}(x)_i \neq f_{\mathcal{X}}(y)_i$. $\square$

The following results are true in dimension 1 only. They are open for higher dimensions.

**Proposition 31.** *In dimension* 1*, surjectivity implies* $\mathfrak{P}$*-surjectivity.*

**Proof.** Let $\mathcal{A}$ be a surjective one-dimensional sand automaton of radius $r$, and $x^0$ a periodic configuration of period $p \in \mathbb{N}$. Let $x$ be a pre-image of $x^0$ by $\mathcal{A}$. We build a periodic configuration $y$ from $x$, whose image is $x^0$. Let $X = \{(x_{k-r}, \ldots, x_{k+r-1}) \mid \exists \alpha \in \mathbb{Z}, k = \alpha p\}$. Since for every $i \in \mathbb{Z}$, $|x_i - x_i^0| \leqslant r$ (as $\lambda$ returns an element of $[\![-r, r]\!]$), and because $x^0$ is $p$-periodic, there are at most $(2r+1)^{2r}$ elements in $X$.

Let $k_1 = \alpha_1 p$ and $k_2 = \alpha_2 p$, $k_1 < k_2$, such that $(x_{k_1-r}, \ldots, x_{k_1+r-1}) = (x_{k_2-r}, \ldots, x_{k_2+r-1})$. Let the $(k_2 - k_1)$-periodic configuration $y$ where the period is defined by (see Fig. 12 for the construction) $y_{k_1+i} = x_{k_1+i}$ for all $0 \leqslant i < k_2 - k_1$. It is easy to see that $f(y) = x^0$, because for every point within the period of $y$, the automaton sees the same neighborhood as for $x$ (due to the construction of $y$), so it acts in the same correct way. And as $k_2 - k_1$ is a multiple of $p$, each period of $y$ coincides with a period of $x^0$, so the image of $y$ is equal to $x$ everywhere: $\mathcal{A}$ is $\mathfrak{P}$-surjective. $\square$

In dimensions greater than 1, the above problem is currently open, we have no direct proof nor counter-example. The problem is due to the fact that in dimension 2 and above, the size of the perimeter of a ball (the $2r$ sequence we used in $X$ for the proof in dimension 1) is linked to the size of the ball. Therefore, we cannot say that there is a finite number of perimeters, and then stick them together to build the periodic configuration.

**Corollary 32.** *In dimension* 1*,* $\mathfrak{F}$*-surjectivity implies* $\mathfrak{P}$*-surjectivity.*

**Proof.** $\mathfrak{F}$-surjectivity implies surjectivity (Proposition 27), which implies $\mathfrak{P}$-surjectivity (Proposition 31) in dimension 1. $\square$

The question whether the above corollary is true in dimension 2 or higher is still open and its solution appears to be quite difficult.

Clearly, injectivity implies $\mathfrak{F}$-injectivity and $\mathfrak{P}$-injectivity, but the converse implications are false. In fact, Proposition 30 shows that $\mathfrak{F}$-injectivity does not imply injectivity. The fact that $\mathfrak{P}$-injectivity does not imply injectivity is proved by the following proposition.

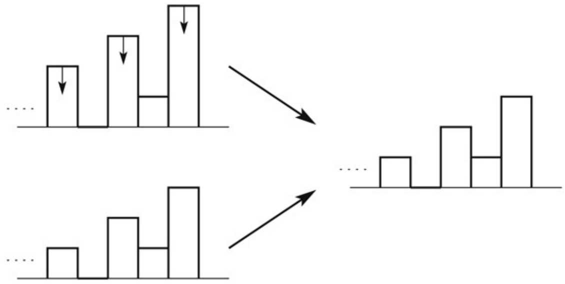


Fig. 13. Examples of the action of $\mathcal{Y}$.

**Proposition 33.** *There is a sand automaton which is $\mathfrak{F}$-injective, $\mathfrak{P}$-injective but not injective.*

**Proof.** Consider the following SA $\mathcal{Y} = \langle 2, \lambda_{\mathcal{Y}} \rangle$, where

$$\forall a, b, c \in \widetilde{[\![-2, 2]\!]}, \quad \begin{array}{l} \lambda_{\mathcal{Y}}(+\infty, a, b, c) = -1, \\ \lambda_{\mathcal{Y}}(2, a, b, c) = -1, \\ \lambda_{\mathcal{Y}}(1, a, b, c) = -1, \\ \lambda_{\mathcal{Y}}(0, a, b, c) = -1, \\ \lambda_{\mathcal{Y}}(-1, -\infty, a, b) = -1, \end{array}$$

and everything else returns 0. Fig. 13 shows two meaningful behaviors of the automaton that will be used in this proof.

Consider the two configurations $x$ and $y$ defined as follows

$$\forall i \in \mathbb{Z}, \qquad \begin{cases} x_{2i} = i \\ x_{2i+1} = i + 2, \end{cases} \quad \begin{cases} y_{2i} = i \\ y_{2i+1} = i + 3. \end{cases}$$

It is not difficult to see that $f_{\mathcal{Y}}(x) = f_{\mathcal{Y}}(y) = x$ (see also Fig. 13). Hence, $\mathcal{Y}$ is not injective. In order to show that $\mathcal{Y}$ is injective over finite and periodic configurations, we need an intermediate result: if $x$, $y$ are two distinct configurations such that $f_{\mathcal{Y}}(x) = f_{\mathcal{Y}}(y)$, then there are infinitely many differences, of infinitely many different values. More precisely, we show that if $x_i > y_i$ then $x_{i-2} > y_{i-2}$ and $x_{i-2} < x_i$.

Assume $x_i \neq y_i$ for some $i$, and let $f_{\mathcal{Y}}(x) = f_{\mathcal{Y}}(y)$. Then, without loss of generality, one can choose $x_i = y_i + 1$ (the difference cannot be greater than one, because $\lambda_{\mathcal{Y}}$ only returns $-1$ or 0). Therefore, a rule which returns 0 is applied to $y$ at position $i$, which means that $y_{i-2} \leqslant y_i - 1$ (since $\lambda_{\mathcal{Y}}(a, -, -, -)$ returns 0 only if $a \leqslant -1$). For the same reason, one of the five rules which returns $-1$ is applied to $x$ at position $i$, hence $x_{i-2} \geqslant x_i - 1$. So it holds that

$$x_{i-2} \geqslant x_i - 1 = y_i \geqslant y_{i-2} + 1 > y_{i-2}. \tag{2}$$

The first consequence of these inequalities is that if there is a difference somewhere, there are infinitely many differences, hence $\mathcal{Y}$ is $\mathfrak{F}$-injective. Indeed, two finite configurations cannot have infinitely many differences, so two different finite configurations have a different image.

Moreover, $x_{i-2} = y_{i-2} + 1$ to ensure $f_{\mathcal{Y}}(x) = f_{\mathcal{Y}}(y)$. So the chain of Inequalities (2) above are in fact equalities; in particular $x_{i-2} = x_i - 1$. Therefore, it holds that $\cdots < x_{i-4} < x_{i-2} < x_i$, which proves that different periodic configurations have different images (a periodic configuration contains a finite number of different columns, which is contradicted by the above inequality). As a consequence, $\mathcal{Y}$ is $\mathfrak{P}$-injective. □

The next three propositions show that, unlike for cellular automata, there are no implications between any of the $\mathfrak{U}$-surjectivity and $\mathfrak{U}$-injectivity properties.

**Proposition 34.** *The* SA $\mathcal{S}$ *is $\mathfrak{U}$-surjective for $\mathfrak{U} = \mathfrak{C}, \mathfrak{F}, \mathfrak{P}$. The SA $\mathcal{S}^r$ is $\mathfrak{U}$-injective for $\mathfrak{U} = \mathfrak{C}, \mathfrak{F}, \mathfrak{P}$.*

**Proof.** It is not difficult to see that $\mathcal{S} \circ \mathcal{S}^r = id$, but $\mathcal{S}^r \circ \mathcal{S} \neq id$ (just use the configuration $x$ defined by $x_0 = 2$ and $x_i = 0$ for $i \in \mathbb{Z} \setminus \{0\}$). The first equation implies that $\mathcal{S}$ is surjective and $\mathcal{S}^r$ is injective. Moreover, since the pre-image by $\mathcal{S}$ of a configuration is computed by $\mathcal{S}^r$, a SA with $\lambda_{\mathcal{S}^r}(0, 0) = 0$, the pre-image of a finite configuration is finite. It is also periodic if the initial configuration was periodic. Hence, we have the first part of the thesis. The second part is a consequence of the injectivity of $\mathcal{S}^r$. □

$$\begin{tikzcd} S_{\mathfrak{P}} \arrow[dd, shift left] & \\ & S_{\mathfrak{F}} \arrow[ul, "1"'] \arrow[dl] \\ S \arrow[uu, shift left, "1"] & \end{tikzcd} \qquad \begin{tikzcd} & I_{\mathfrak{P}} \arrow[dd] \\ I \arrow[ur] \arrow[dr] & \\ & I_{\mathfrak{F}} \end{tikzcd}$$

Fig. 14. Relations between basic set properties for sand automata. $I$ means injectivity and $S$ surjectivity. $I_{\mathfrak{U}}$ (resp. $S_{\mathfrak{U}}$) means injectivity (resp. surjectivity) restricted to $\mathfrak{U}$ Arrows indicate implications, the symbol $\xrightarrow{1}$ means that the implication is true in dimension 1 and open in higher dimensions. When there is no arrow, the implication is false.

**Proposition 35.** *The* SA $\mathcal{S}$ *is not* $\mathfrak{U}$*-injective for* $\mathfrak{U} = \mathfrak{C}, \mathfrak{F}, \mathfrak{P}$.

**Proof.** Consider the following finite configurations $x, y$ where $x_i = 0$ for $i \in \mathbb{Z}$, $y_i = 0$ for $i \in \mathbb{Z} \setminus \{0, 1\}$, $y_0 = 1$, and $y_1 = -1$. Clearly, $f_{\mathcal{S}}(x) = f_{\mathcal{S}}(y) = x$. Now, consider the periodic configuration $z$ with $z_{2i} = 1$ and $z_{2i+1} = -1$ for every $i \in \mathbb{Z}$, again $f_{\mathcal{S}}(x) = f_{\mathcal{S}}(z) = x$. $\square$

**Proposition 36.** *The* SA $\mathcal{S}^r$ *is not* $\mathfrak{U}$*-surjective for* $\mathfrak{U} = \mathfrak{C}, \mathfrak{F}, \mathfrak{P}$.

**Proof.** Consider the following finite configuration $x$, where $x_0 = 2$ and $x_i = 0$ if $i \neq 0$. Assume that $x$ has a pre-image $y$. There are only three possibilities for the value of $y_0$:

$y_0 = 3$ : then the local rule has to return $-1$, which implies that $y_{-1} \geqslant 5$. But $f_{\mathcal{S}^r}(y)_{-1} = 0$, this value cannot be reached from 5;

$y_0 = 2$ : the column is unchanged, which means that ($y_{-1} \leqslant 3$ or $y_1 \leqslant 0$) and ($y_{-1} \geqslant 4$ or $y_1 \geqslant 1$). For the same reason as before, $y_{-1}$ cannot be greater than 4, hence $y_1 \geqslant 1$. This means that the local rule applied at position 1 returns $-1$, in other words that $y_0 \geqslant 3$, which contradicts the first hypothesis;

$y_0 = 1$ : $\lambda_{\mathcal{S}^r}$ returns $+1$, so $y_1 \leqslant -1$. Hence, at position 1 $\lambda_{\mathcal{S}^r}$ also returns $+1$. That means, in particular, that $y_2 \leqslant -3$, which is impossible if one has to obtain $f_{\mathcal{S}^r}(y)_2 = 0$.

We have found a finite configuration with no pre-image. This means that $\mathcal{S}^r$ is not surjective both on $\mathfrak{C}$ and on $\mathfrak{F}$. To show that $\mathcal{S}^r$ is not $\mathfrak{P}$-surjective, one can consider the configuration $x$ where $x_{4i+1} = 2$ for every $i \in \mathbb{Z}$, and everywhere else $x_k = 0$. The proof is similar to the previous part, since the 4 elements of the period act as if the configuration was finite (the radius is 1, so they do not "see" farther than one column ahead and one column back). $\square$

The results about basic set properties are summarized on Fig. 14.

If one compares these relations to the similar properties for cellular automata [15] (see Fig. 15), one remarks that in the latter case there exist many links between surjectivity and injectivity. The lack of relations for sand automata confirms that the two systems have different dynamics, and suggests that studying the decidability of these properties might be difficult.

## 7. Grain conserving sand automata

The notion of *grain conserving* (GC) sand automata is very similar to the one of number conserving cellular automata [18–20]. Roughly, a SA is said to be GC if it does not create nor destroy grains in a configuration (which is a reasonable constraint for typical sandpile models). In this section, we will define precisely GC SA, and show that this property is decidable, in a way similar to what was done in [19].

To be able to compute the number of grains of a configuration, we need to restrict to specific configurations, limited in height and width.

**Definition 37** (*Finite Grain Conserving*). A SA $\mathcal{A}$ of global function $f$ is said to be *finite grain conserving* (FGC) if

$$\forall x \in \mathfrak{F}, \quad \sum_{i \in \mathbb{Z}^d} x_i = \sum_{i \in \mathbb{Z}^d} f(x)_i .$$

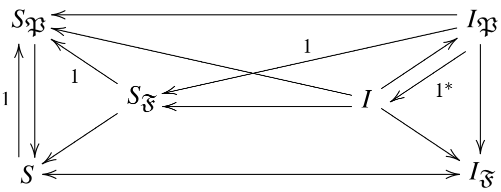



Fig. 15. Relations between basic set properties for cellular automata. The symbols have the same meaning as on Fig. 14, and the symbol $\xrightarrow{1^*}$ means that the implication is true in dimension 1 and false in higher dimensions. These results are taken from [15].

**Definition 38** (*Periodic Grain Conserving*). A SA $\mathcal{A}$ of global function $f$ is said to be *periodic grain conserving* (PGC) if

$$\forall x \in \mathfrak{P}, \qquad \sum_{\substack{i \in \mathbb{Z}^d \\ 0 \preccurlyeq i \prec p}} x_i = \sum_{\substack{i \in \mathbb{Z}^d \\ 0 \preccurlyeq i \prec p}} f(x)_i,$$

where $p \in \mathbb{N}^d$ is the period of $x$ (the grains are counted over the period).

The following theorem removes the ambiguity about these two definitions, showing that they are equivalent. For this reason we simply write GC from now on.

**Theorem 39.** *The definitions PGC and FGC are equivalent.*

**Proof.** We prove that FGC implies PGC. Let $\mathcal{A}$ be a FGC SA of global rule $f$, of radius $r$. Suppose that $\mathcal{A}$ is not PGC, *i.e.* there is a configuration $x \in \mathfrak{P}$ of period $p = (p_1, \ldots, p_d) \in \mathbb{N}^d$ such that $\sum_{0 \preccurlyeq i \prec p} x_i = k \neq k' = \sum_{0 \preccurlyeq i \prec p} f(x)_i$. From $x$ we build a finite configuration whose grain content is not preserved, contradicting the fact that $\mathcal{A}$ is FGC.

Let $x^p$ be the matrix of all $x_i$'s, for $i \in \mathbb{Z}^d$, $0 \preccurlyeq i \prec p$. Let $y \in \mathfrak{F}$ be such that it contains $x^p$ $\alpha$ times in every dimension. On the borders, $y$ is filled with the $r$ values of $x^p$ that would have been there if $y$ was periodic (see Fig. 16 for the construction in dimension 2).

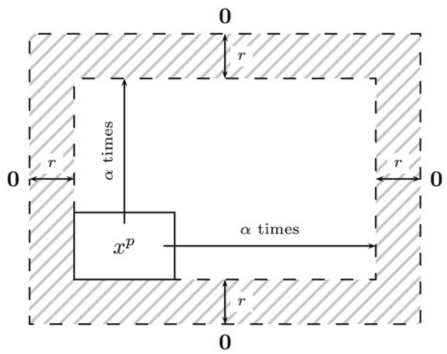



Fig. 16. Construction of the finite configuration $y$ starting from $x$, in dimension 2.

Let $M = \max_{0 \preccurlyeq i \prec p} \max(x_i, f(x)_i)$ and $m = \min_{0 \preccurlyeq i \prec p} \min(x_i, f(x)_i)$ be the two extremum values in both $x$ and $f(x)$. Counting the grains in $y$, it holds that

$$k\alpha^d + P(\alpha)m \leqslant \sum_{i \in \mathbb{Z}^d} y_i \leqslant k\alpha^d + P(\alpha)M,$$

where $P(\alpha) = \prod_{j=1}^d (\alpha p_j + 2r) - \prod_{j=1}^d (\alpha p_j)$ is the number of hatched elements. Note that $P$ is polynomial, of degree $d-1$. Similarly,

$$k'\alpha^d + Q(\alpha)m \leqslant \sum_{i \in \mathbb{Z}^d} f(y)_i \leqslant k'\alpha^d + Q(\alpha)M,$$

where $Q(\alpha) = \prod_{j=1}^d (\alpha p_j + 4r) - \prod_{j=1}^d (\alpha p_j)$ is the number of non-zero elements in $f(y)$, apart from the center part. $Q$ is also polynomial, of degree $d-1$.

If $k > k'$, because $P$ and $Q$ are of degree $d-1$, it is possible to choose $\alpha$ big enough so that

$$\sum_{i\in\mathbb{Z}^d} y_i \geqslant k\alpha^d + P(\alpha)m > k'\alpha^d + Q(\alpha)M \geqslant \sum_{i\in\mathbb{Z}^d} f(y)_i.$$

In a similar way, if $k < k'$ and $\alpha$ is big enough then

$$\sum_{i\in\mathbb{Z}^d} f(y)_i \geqslant k'\alpha^d + Q(\alpha)m > k\alpha^d + P(\alpha)M \geqslant \sum_{i\in\mathbb{Z}^d} y_i.$$

In both cases, there is a contradiction with the fact that $\mathcal{A}$ is FGC, therefore $\mathcal{A}$ has to be PGC.

Now we prove that PGC implies FGC. Let $\mathcal{A}$ be a PGC SA of radius $r$, of global rule $f$. Let $x \in \mathfrak{F}$, construct $y \in \mathfrak{P}$ as in the proof of Proposition 29 (Fig. 10, page 14, in dimension 2): let $l \in \mathbb{N}$ be such that $2l+1 > |x|$, then set $y_i = x_i$ for $|i| < |l|$, and $y_i = 0$ for $|l| \leqslant |i| < |l|+r$. By construction, for all $|i| < |l|+r$, $f(y)_i = f(x)_i$ because the neighborhood is identical. Therefore,

$$\begin{aligned}\sum_{i\in\mathbb{Z}} f(x)_i = \sum_{0\leqslant|i|<|l|+r} f(x)_i = \sum_{0\leqslant|i|<|l|+r} f(y)_i &= \sum_{0\leqslant|i|<|l|+r} y_i \quad (\mathcal{A}\text{ is PGC})\\ &= \sum_{0\leqslant|i|<|l|+r} x_i = \sum_{i\in\mathbb{Z}} x_i. \quad \square\end{aligned}$$

In the same way as it was done in [19], the fact that an automaton is GC can be decided, provided that the local rules satisfy specific conditions. First we exhibit these conditions for the simplest case, automata of radius 1 in dimension 1, then we give the general formula.

To "simplify" the notations and the calculi, we introduce for every automaton $\mathcal{A} \equiv \langle r, \lambda\rangle$ the function $\gamma : \widetilde{\mathbb{Z}}^{[\![-r,r]\!]^d} \mapsto \widehat{[\![-r,r]\!]}$, defined by

$$\forall x \in \mathfrak{C}, \forall i \in \mathbb{Z}^d, \quad \gamma(M_r^i(x)) = \lambda(R_r^i(x)),$$

where $M_r^i(x)$ is the matrix containing all elements $x_j$ such that $|j-i| \leqslant r$. For example in dimension 1, radius 1, one has

$$\gamma(2,3,3) = \gamma(-3,-2,-2) = \lambda(-1,0).$$

**Proposition 40.** *In dimension* 1, *a* SA $\mathcal{A} \equiv \langle 1, \lambda\rangle$ *is GC if and only if for all* $a, b, c \in \mathbb{Z}$,

$$\gamma(a,b,c) = \gamma(0,0,b) - \gamma(0,0,a) + \gamma(0,b,c) - \gamma(0,a,b).$$

**Proof.** Let $\mathcal{A} \equiv \langle 1, \lambda\rangle$ be a GC SA of global rule $f$. Let $a, b, c \in \mathbb{Z}$, and $x = (\ldots, 0, a, b, c, 0, \ldots) \in \mathfrak{F}$.

$\mathcal{A}$ is in particular FGC (Theorem 39), hence $\sum_{i\in\mathbb{Z}} f(x)_i = \sum_{i\in\mathbb{Z}} x_i = a+b+c$. When we apply $f$ to $x$, it also holds

$$\sum_{i\in\mathbb{Z}} f(x)_i = \gamma(0,0,a) + \gamma(0,a,b) + a + \gamma(a,b,c) + b + \gamma(b,c,0) + c + \gamma(c,0,0)$$

and hence

$$\gamma(a,b,c) = -\gamma(0,0,a) - \gamma(0,a,b) - \gamma(b,c,0) - \gamma(c,0,0). \tag{3}$$

To get rid of the $\gamma(b,c,0)$ term, we perform the same operations on the finite configuration $y = (\ldots, 0, b, c, 0, \ldots)$ (*i.e.* we remove the first element of $x$), which leads to

$$\gamma(b,c,0) = -\gamma(0,0,b) - \gamma(0,b,c) - \gamma(c,0,0).$$

By injecting this result in Eq. (3), we obtain the final condition.

For the converse implication, let $\mathcal{A}$ be a SA which satisfies this condition, and let $x \in \mathfrak{P}$ of period $p$. $x = (\ldots, x_p, x_1, x_2, \ldots, x_p, x_1, \ldots)$. It holds that

$$\sum_{i=1}^{p} f(x)_i = x_1 + \gamma(x_p, x_1, x_2) + x_2 + \gamma(x_1, x_2, x_3) + \cdots + x_p + \gamma(x_{p-1}, x_p, x_1).$$

When replacing the $\gamma$'s by the sums, all the terms simplify and it remains only $\sum_{i=1}^{p} f(x)_i = x_1 + \cdots + x_p$, hence $\mathcal{A}$ is GC. □

Every time a dimension is added, the result becomes a little bit trickier, and formulas much heavier. We give an sketch of the proof of the corresponding condition for dimension 2.

**Proposition 41.** *A* SA *in dimension* 2 *is GC if and only if the following formula holds for all* $(x_{i,j}) \in \mathbb{Z}^{[\![-r,r]\!]^2}$.

$$\gamma\begin{pmatrix} x_{-r,-r} & \cdots & x_{r,-r} \\ \vdots & \ddots & \vdots \\ x_{-r,r} & \cdots & x_{r,r}\end{pmatrix} = -\sum_{\substack{i=0\\ i+j>0}}^{2r}\sum_{j=0}^{2r}\gamma\begin{pmatrix} \mathbb{O}_{i,j} & & 0 & \\ & x_{-r,-r} & \cdots & x_{r-i,-r} \\ 0 & \vdots & \ddots & \vdots \\ & x_{-r,r-j} & \cdots & x_{r-i,r-j}\end{pmatrix}$$
$$+\sum_{i=1}^{2r}\sum_{j=0}^{2r}\gamma\begin{pmatrix} \mathbb{O}_{i,j} & & 0 & \\ & x_{-r+1,-r} & \cdots & x_{r+1-i,-r} \\ 0 & \vdots & \ddots & \vdots \\ & x_{-r+1,r-j} & \cdots & x_{r+1-i,r-j}\end{pmatrix}$$
$$+\sum_{i=0}^{2r}\sum_{j=1}^{2r}\gamma\begin{pmatrix} \mathbb{O}_{i,j} & & 0 & \\ & x_{-r,-r+1} & \cdots & x_{r-i,-r+1} \\ 0 & \vdots & \ddots & \vdots \\ & x_{-r,r+1-j} & \cdots & x_{r-i,r+1-j}\end{pmatrix}$$
$$-\sum_{i=1}^{2r}\sum_{j=1}^{2r}\gamma\begin{pmatrix} \mathbb{O}_{i,j} & & 0 & \\ & x_{-r+1,-r+1} & \cdots & x_{r+1-i,-r+1} \\ 0 & \vdots & \ddots & \vdots \\ & x_{-r+1,r+1-j} & \cdots & x_{r+1-i,r+1-j}\end{pmatrix}.$$

*In this formula,* $\mathbb{O}_{i,j}$ *is the matrix with* $i$ *columns and* $j$ *lines containing* 0 *everywhere.*

**Sketch of the proof.** Consider a GC SA $\mathcal{A}$ of global rule $f$, and the finite configuration $x$ containing $X = \begin{pmatrix} x_{-r,-r} & \cdots & x_{r,-r} \\ \vdots & \ddots & \vdots \\ x_{-r,r} & \cdots & x_{r,r}\end{pmatrix}$ at the center, and 0 elsewhere. Counting the grains in $x$ and $f(x)$ gives a expression of $\gamma(X)$. To remove the terms which do not have a 0 at position $(-r, -r)$ (top-left), we repeat these operations on the two finite configurations $y$ and $z$ which are like $x$ without the first column for $y$, and without the first line for $z$. Finally repeat this on the configuration $t$ which is $x$ without both the first line and the first column, you get the result.

Conversely if $\mathcal{A}$ satisfies the formula, when summing over the period of the image of a periodic configuration, it is evident that all terms disappear, and that the number of grains is preserved. □

To give the general formula in dimension $d$, radius $r$, we introduce for every automaton $\mathcal{A}$, for every matrix $(x) \in \mathbb{Z}^{[\![-r,r]\!]^d}$, a new function $g : \{0,1\}^d \mapsto \mathbb{Z}$ defined by

$$g(a_1,\ldots,a_d) = \sum_{\substack{k_1=a_1\\ k_1+\cdots+k_d>0}}^{2r} \cdots \sum_{k_d=a_d}^{2r} \gamma\big(M(k_1,\ldots,k_d)\big),$$

where $M(k_1,\ldots,k_d)$ is the $d$-dimensional matrix defined by

$$M(k_1,\ldots,k_d)_{i_1,\ldots,i_d} = \begin{cases} 0 & \text{if } i_1 \leqslant k_1 \text{ or } \cdots \text{ or } i_d \leqslant k_d, \\ x_{-r+a_1+i_1-k_1-1,\ldots,-r+a_d+i_d-k_d-1} & \text{otherwise,}\end{cases}$$

in other words for every dimension $i$, $M$ contains zeros in the first $k_i$ elements then it begins with $x_{\dots,-r+a_i,\dots}$. For example in dimension 1,

$$g(a) = \sum_{k=1}^{2r} \gamma(\underbrace{0,\dots,0}_{k \text{ times}}, x_{-r+a}, \dots, x_{r+a-k}).$$

In dimension 2, it looks like the double sums in Proposition 41.

**Theorem 42.** *A* SA $\mathcal{A} = \langle r, \lambda\rangle$ *in dimension* $d$ *is GC if and only if for all matrices* $\mathbf{M} = (x_{i_1,\dots,i_d}) \in \mathbb{Z}^{[\![-r,r]\!]^d}$,

$$\gamma(\mathbf{M}) = \sum_{a_1,\dots,a_d=0}^{1} (-1)^{a_1+\dots+a_d+1} g(a_1,\dots,a_d).$$

**Sketch of the proof.** This proof is very similar to the ones of the previous propositions. If a SA is GC, then we count the grains on the finite configuration with 0 everywhere and **M** inside. There are as many grains as in its image by $f$. This gives a first equation.

Now the same operation can be performed on every configuration which contains **M** without every possible combination of first rows in every dimension (at first, only one row is removed, then two rows, until $d$ rows are removed). This corresponds to the $a_i$'s, $a_i = 1$ means that the first row in dimension $i$ is removed. When inserting all these equalities in the first one, we obtain the result.

Conversely, if $f$ satisfies this formula then when we make the sum over the period of any configuration, all terms are cancelled by another one. □

**Corollary 43.** *The conservation of grains for any* SA *is decidable.*

**Proof.** It suffices to check the conditions of Theorem 42 for a finite number of values. Indeed, to build every possible neighborhood, it is sufficient to fix the central element $x_{0,\dots,0}$ at 0, and to try every possible values in $[\![-r, r]\!]$ for all other elements. Moreover, in order to have every possible neighborhood in the right part of the equality (the expressions with $g$), we need to ensure that the difference between any two columns belong to $[\![-r, r]\!]$. This is necessary because every column will be at some time the reference point of a matrix $M$.

The number of conditions that have to be verified is exponential, but finite: it is sufficient to choose the first element in $[\![-r-1, r+1]\!]$, the second in $[\![-2r-2, 2r+2]\!]$, and so on until the $[(2r+1)^d - 1]$th element. This means "no more" than $\prod_{i=1}^{(2r+1)^d-1} i(2r+3)$ tests. □

## 8. Ultimate periodicity

Understanding the dynamical behavior of SA seems very difficult. This is confirmed by the main result of this section: ultimate periodicity, one of the simplest dynamical behaviors, is undecidable for sand automata. This section explicits the sketch of proof which can be found in [2].

Given a SA $f$, a configuration $x$ is ultimately periodic if $\exists p, t \in \mathbb{N}$ such that $\forall i, k \in \mathbb{N}$, $f^{pk+i+t}(x) = f^{i+t}(x)$. A SA $f$ is $\mathfrak{U}$-ultimately periodic if for all $x \in \mathfrak{U}$, $x$ is ultimately periodic for $f$.

**Problem ULT**($\mathfrak{U}$)

INSTANCE: a SA $\mathcal{A} = \langle \lambda, r\rangle$;

QUESTION: is every configuration in $\mathfrak{U}$ ultimately periodic for $\mathcal{A}$?

We reduce the problem of the ultimate periodicity of a sand automaton to the halting problem of a two registers machine with finite control, started with both registers at 0. In the following subsections we explicit how the simulation of such a machine by a sand automaton is done.

### 8.1. Construction of the automaton

The reduction will be made from a two registers machine $\mathcal{M}$ defined by $\mathcal{M} = \langle Q, q_0, q_f, \delta\rangle$, where $Q$ is a finite set of states, $q_0 \in Q$ is the initial state, $q_f \in Q$ the final state. The registers $R_1$ and $R_2$ always contain positive integer values. In our case, $\mathcal{M}$ is always started with both registers at 0.

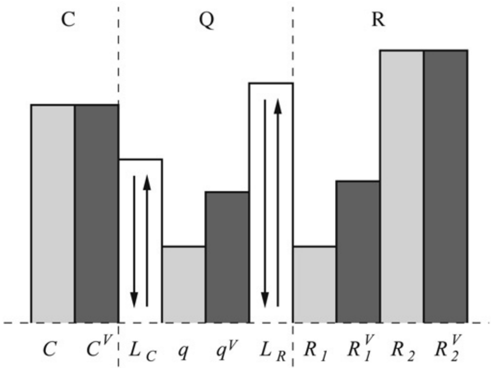



Fig. 17. Simulation of a two registers machine by a SA.

The function $\delta : Q \times \{0, 1\} \times \{0, 1\} \mapsto Q \times \{1, 2\} \times \{-1, 0, +1\}$ is the *transition function*. The second and third arguments of $\delta$ indicate whether or not the registers are 0 (hence a 1 means that the register contains a value strictly greater than 0). $\delta$ returns the new state, the number of the register which is modified (1 or 2), and its modification (decrease by 1, no change, increase by 1). For clarity, we denote these transitions by the expression $\delta(q, b_1, b_2) = (q', R_i + j)$.

For example, the rules

$$\begin{cases} \delta(q_0, 0, 0) & = & (q_1, R_1 + 1) \\ \delta(q_1, 0, -) & = & (q_f, R_1 + 0) \\ \delta(q_1, 1, -) & = & (q_2, R_1 - 1) \\ \delta(q_2, -, -) & = & (q_3, R_2 + 1) \\ \delta(q_3, -, -) & = & (q_1, R_2 + 1) \end{cases}$$

define a machine which first initializes $R_1$ to 1, then multiplies it by 2 and puts the result in $R_2$.

A two registers machine $\mathcal{M}$ (started with both registers at 0) is associated with a SA $\mathcal{S}_{\mathcal{M}}$. Fig. 17 illustrates the general "architecture" of $\mathcal{S}_{\mathcal{M}}$: the idea is that $\mathcal{S}_{\mathcal{M}}$ uses a certain number of grain stacks for the registers (R) and for the finite control (Q) in order to simulate the iterations of $\mathcal{M}$. For technical reasons we also need a counter (C) which counts the number of iterations of $\mathcal{M}$.

In order to describe more precisely $\mathcal{S}_{\mathcal{M}}$ we need the following "tips and tricks" which are fundamental in the construction.

**The lifts**. The control has to send commands both to the registers (R) and to the counters (C). The point is that the radius of the local rule is finite and the difference of height between the control and the registers or the counters could be much bigger than the radius. Hence, the control cannot deliver commands directly to the registers or to the counters. This problem can be solved by introducing two more columns which we call *lifts* : $L_C$ delivers commands to the counters and $L_R$ delivers commands to the registers (see Fig. 17).

**Knowing themselves**. The local rule of $\mathcal{S}_{\mathcal{M}}$ is formed by several sub-rules. Each sub-rule concerns the evolution of a single column of the simulation zone of $\mathcal{S}_{\mathcal{M}}$. The point is that each column must know "which it is" in order to apply the right sub-rule. This problem is solved by splitting each column $a$ into two columns $(a_l, a_r)$ and the "identity" of the original column is coded by the difference of height between $a_l$ and $a_r$. For example, a difference of 1 says that $a$ is the counter $C$, 2 stands for $C^V$ and so on. We also use a height difference to code an error symbol $E$ whose meaning will be explained later.

In the following, when speaking of height of a column $a = (a_l, a_r)$ we will always mean the height of $a_r$ since $a_l$ is simply the height of $a_r$ plus the "identity" number.

Finally, the height of $q_r$ is used as (relative) zero height by all other columns when needed.

**Commands, colors and states**. The idea of coding "identity" information in the difference between pairs of successive columns can be used to store additional information which will be useful for the simulation. For example, one can code the following commands for the lifts: $C_{+1}$ which increases the counter $C$ by one; $C^V_{\to 0}$ that resets $C^V$; $R_{1,-1}$ which decreases $R_1$ by one; $L_{\searrow}$ which instructs the lift to go down and so on. We also use height differences for coloring lifts ($\mathbf{S}$, $\mathbf{V}_0$, $\mathbf{V}$, $\mathbf{C}$). The use and meaning of commands and colors will be detailed in Section 8.2.

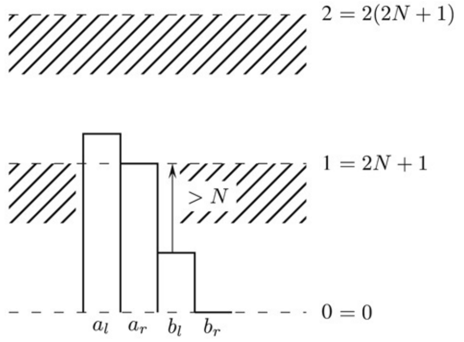



Fig. 18. How to distinguish “the left side from the right side”.

Finally, we need to code the state of the control $q$ (or $q^V$). Once more, this piece of information can be coded into the height difference.

**Removing ambiguities**. Let $N$ be the biggest difference used to code objects (or actions, see above) needed in the simulation. In the following, in order to maintain a strict correspondence between the two registers machine and the simulated model, we prefer to say that “a column $a = (a_l, a_r)$ is increased by $t \in \mathbb{N}$” even if in reality in $\mathcal{S}_{\mathcal{M}}$, $a_r$ is increased by $(2N + 1) \cdot t$ and $a_l$ is increased by $(2N + 1) \cdot t + \alpha$ where $\alpha \in \{-N + 1, \ldots, 0, \ldots, N - 1\}$ is meant to code the modification of the state of the column or its color. This trick avoids ambiguities in the “identity” of the columns as can be seen in Fig. 18. All rightmost columns with a $_r$ subscript are located at levels $k \cdot (2N + 1)$, while the leftmost columns avoid the cross-hatched zone and remain between the line $k \cdot (2N + 1)$ and $k \cdot (2N + 1) + N$. As a consequence, the difference between any consecutive $a_r$ and $b_l$ exceeds $N$ and cannot be mistaken for a code: $a_r$ is guaranteed to be the right column of a pair, and $b_l$ the left column of another.

### 8.2. Simulation

Each iteration of $\mathcal{M}$ can be simulated by $\mathcal{S}_{\mathcal{M}}$ in three main steps:

**S**. *simulation* of one iteration of $\mathcal{M}$;

**V**. *verification* from the beginning to the current iteration, in the verification columns (those with a $^V$ superscript);

**C**. *comparison* between the results of the first two steps, to ensure that the simulation is correct.

These three steps are necessary since not all initial configurations of $\mathcal{S}_{\mathcal{M}}$ represent valid computations of $\mathcal{M}$. For this reason, $\mathcal{S}_{\mathcal{M}}$ is equipped with a verification part that is able to simulate $\mathcal{M}$ when started with both registers at zero. Then $\mathcal{S}_{\mathcal{M}}$ compares the current state with the one obtained in the verification part. If they coincide, the counter (C) is increased by one and a new iteration of $\mathcal{M}$ is simulated; otherwise $\mathcal{S}_{\mathcal{M}}$ evolves to a periodic configuration.

In the following, lifts are colored according to the current simulation step (**S**, **V**$_0$, **V**, **C**). The following paragraphs explain in detail all these steps, giving examples of local rules.

**The beginning**. At the beginning of the simulation, $C$ contains the number of simulation steps (w.r.t. $\mathcal{M}$) since the beginning, $q = (q_l, q_r)$ contains the current state of $\mathcal{M}$, the registers $R_1$ and $R_2$ contain some value. The lifts $L_C$ and $L_R$ are at 0 (relatively to $q_r$). Moreover, the lifts are in color **S**.

All other columns contain arbitrary values. They will be reset later on when necessary.

**S**. *Simulation step*

In this step, $\mathcal{S}_{\mathcal{M}}$ simulates a single iteration of $\mathcal{M}$. For example, assume that $R_1$ and $R_2$ contain a strictly positive value and that $\delta(q_1, 1, 1) = (q_2, R_i + j)$ for some $i \in \{1, 2\}$ and $j \in \{-1, 0, +1\}$. Then, $\mathcal{S}_{\mathcal{M}}$ changes $q_1$ into $q_2$ in column $q$ and at the same time fires $L_C$ with the command $C_{+1}$ and $L_R$ with the command $R_{i,j}$. Below we give the local rules of $\mathcal{S}_{\mathcal{M}}$ which perform the update of $q$ for this transition.

$$\begin{cases}\lambda\Big(\ldots,\mathbf{L_C},\alpha_{q_1}\ \Big|\ \underbrace{-,-}_{q^V},\mathbf{L_R},\mathbf{R_1},\underbrace{-,-}_{R_1^V},\mathbf{R_2},\ldots\Big)=0 & \text{(right)}\\ \lambda\Big(\ldots,\mathbf{L'_C}\ \Big|\ -\alpha_{q_1},\overbrace{-,-},\mathbf{L'_R},\mathbf{R'_1},\overbrace{-,-},\mathbf{R'_2},\ldots\Big)=\alpha_{q_2}-\alpha_{q_1} & \text{(left)}\end{cases}$$

with

$$\mathbf{L_C}=\underbrace{\alpha_{L_C,\mathbf{S}},0}_{L_C\text{ at }0,\ \mathbf{S}\text{-colored}},\qquad \mathbf{L_R}=\underbrace{\alpha_{L_R,\mathbf{S}},0}_{L_R\text{ at }0,\ \mathbf{S}\text{-colored}},$$
$$\mathbf{L'_C}=\overbrace{\alpha_{L_C,\mathbf{S}}-\alpha_{q_1},-\alpha_{q_1}},\qquad \mathbf{L'_R}=\overbrace{\alpha_{L_R,\mathbf{S}}-\alpha_{q_1},-\alpha_{q_1}},$$

and

$$\mathbf{R_1}=\underbrace{>\alpha_{R_1},>0}_{R_1\neq 0},\qquad \mathbf{R_2}=\underbrace{>\alpha_{R_2},>0}_{R_2\neq 0},$$
$$\mathbf{R'_1}=\overbrace{>(\alpha_{R_1}-\alpha_{q_1}),>-\alpha_{q_1}},\qquad \mathbf{R'_2}=\overbrace{>(\alpha_{R_2}-\alpha_{q_1}),>-\alpha_{q_1}},$$

where $0<\alpha_a\leqslant N$ represents the difference used to code all the characteristics of column $a$ (identity, state, color, *etc.*). In the above formulas, the notation $>x$ means any number greater than $x$, while $-$ means any number. Moreover, the | symbol is used as a delimiter between the neighborhood on the left and on the right.

Surely, the reader has remarked how involved are the formulas for the local rule of $\mathcal{S}_{\mathcal{M}}$. For this reason we prefer to describe them by words in the following. We stress that translating the descriptions into rules is not difficult.

The next iterations are for the lifts to reach their destination height and deliver the command. As a result, $C$ finally increases by 1 and if necessary one of the registers can also have its value modified. Then, $L_C$ and $L_R$ go down (this can be done by turning into the command $L_{\searrow}$), changing their color to $\mathbf{V}_0$.

The step $\mathbf{S}$ ends when both lifts have reached the reference height, and are colored in $\mathbf{V}_0$.

$\mathbf{V}_0$. *Initialization of the verification step*

Before starting the verification step, one should reset the verification columns (*i.e.* those with the $^V$ superscript in Fig. 17). In $\mathcal{S}_{\mathcal{M}}$, it is performed by sending $C^V_{\to 0}$ command to $L_C$ and $R^V_{\to 0}$ to $L_R$. Finally, $q^V$ is set to $q_0$.

The $R^V_{\to 0}$ command starts a sequence of actions. First, $L_R$ goes up until it is above both registers. Then it goes down, forcing the registers to go down with it. The same holds for $C^V_{\to 0}$.

Finally, when the lifts reach the reference height (*i.e.* the height of $q_r$), they turn into color $\mathbf{V}$ to indicate that the initialization step is complete, and that the verification step can begin.

$\mathbf{V}$. *Verification step*

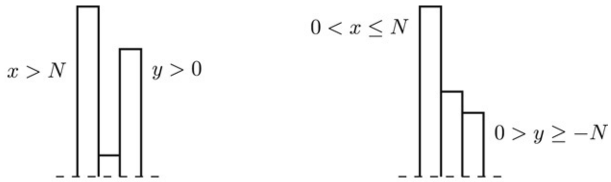
Each time both lifts are on the ground, colored in $\mathbf{V}$, $C$ iterations of $\mathcal{M}$ (started with both registers at 0) are performed in the verification columns. This is done exactly like in step $\mathbf{S}$: the lifts $L_C$ and $L_R$ deliver commands to the counter $C^V$ and to the registers $R_i^V$ ($i\in\{1,2\}$), while the current state $q^V$ is modified according to the rules of $\mathcal{M}$.

Moreover, $L_C$ has to detect when $C=C^V$, which corresponds to the end of the verification step. In that case, it goes down with color $\mathbf{C}$. When it reaches height 0 (*i.e.* the height of $q_r$), $L_R$ checks the color of $L_C$ and turns into the same color. At this point, $C=C^V$, $q$, $R_1$, $R_2$ should be equal to $q^V$, $R_1^V$, $R_2^V$ (the next step will determine if this is really the case), and $L_C$ and $L_R$ are at the reference height colored $\mathbf{C}$.

$\mathbf{C}$. *Comparison step*

The lift $L_C$ is launched and it goes up until it reaches the highest among $C$, $C^V$, $R_1$, $R_1^V$, $R_2$, $R_2^V$. Then, it starts going down, comparing columns two by two when it reaches their height.

If everything is correct *i.e.* $L_C$ reaches 0, then it changes its color into $\mathbf{S}$. At this point $L_R$ become $\mathbf{S}$-colored also and the comparison step is finished.




(a) Column coding for nothing. (b) Identity ambiguity.

Fig. 19. Typical *identity* errors.

If $L_C$ finds that the comparison failed, it changes into the error state $E$, and does not move anymore: the simulation is blocked forever, since all other columns are waiting for $L_C$ to go down. Remark that in this last case, $\mathcal{S}_\mathcal{M}$ is in an ultimately periodic point.
**Terminating the simulation**. For all neighborhoods that were not considered above, the local rule of $\mathcal{S}_\mathcal{M}$ returns 0. This assumption is essential for several proofs that will follow.

*8.3. Halting on errors*

In the following, a configuration of $\mathcal{S}_\mathcal{M}$ is *valid* if it represents a computation of $\mathcal{M}$ when started with both registers at 0. A configuration is *malformed* if it does not respect the "architecture" of $\mathcal{S}_\mathcal{M}$ *i.e.*, for example, the value of the counter is negative *etc.*

If $\mathcal{M}$ halts when started with both registers at 0, then $\mathcal{S}_\mathcal{M}$ evolves to a periodic point when started with a valid configuration. In fact, when the control of $\mathcal{S}_\mathcal{M}$ reaches a halting state, all the other parts freeze in the current value. The point of this section is to show (possibly by adding new local rules) that $\mathcal{S}_\mathcal{M}$ is ultimately periodic also when started from a malformed configuration.

There are two main categories of errors for a particular column. First, *neighborhood* errors which are not due to the column itself, but to its global situation. For example, a pair of columns which code a register, but containing a negative value. Or any misplaced pair of columns, such as 2 state columns in the same configuration. Another neighborhood error is when two consecutive columns code for the error symbol.

Second, when a pair of consecutive columns does not code for anything, or there is an ambiguity in the coding, the configuration is also invalid. This is called an *identity* error.
**Neighborhood errors**. When this type of error occurs one has to prevent any further movement. When a pair of columns finds unexpected values in its neighborhood it changes into the error symbol $E$. In terms of the local rule, this means that any sub-rule concerning a particular type of column $a = (a_l, a_r)$ with an incorrect neighborhood returns 0 for column $a_r$, and for $a_l$ it returns the height difference coding $E$ minus the current identity number.
**Identity errors**. For identity errors the local rule returns 0. This concerns both columns whose neighbors do not code for anything (in this case we have $\lambda(\ldots, x \mid y, \ldots) = 0$, with $x > N$ or $x \leqslant 0$, and $y < -N$ or $y \geqslant 0$, see Fig. 19(a)) and columns which cannot decide which column they are paired with (in this case we have $\lambda(\ldots, x \mid y, \ldots) = 0$, with $0 < x \leqslant N$ and $0 \geqslant y > -N$, see Fig. 19(b)).

From now on, fix a two registers machine $\mathcal{M}$ and let $\mathcal{S}_\mathcal{M}$ be the associated SA given by the above construction. Let $f$ be the global rule of $\mathcal{S}_\mathcal{M}$.

**Lemma 44.** *For any configuration $c \in \mathfrak{F}$ and for any $t \in \mathbb{N}$, $\left|f^t(c)\right| \leqslant |c| + 1$.*

**Proof.** Let $c \in \mathfrak{F}$ and $i$ be its leftmost non-zero value. By construction, we have that $\lambda(-, \ldots, -, 0 \mid -, \ldots, -) = 0$ and hence $\forall j \in \mathbb{Z}\ \forall t \in \mathbb{N}$, $j < i \Rightarrow f^t(c)_j = 0$.

Now, let $k$ be the rightmost non-zero value of $c$. Remark that $f^t(c)_{k+1}$ is always a multiple of $2N + 1$. Since $c_{k+1}$ is either an identity error or the right column of a pair coding for something then, by construction, either it does not increase at all (in the case of an identity error) or it changes by multiples of $2N + 1$. Again, by construction, for any $j > k + 1$, one finds $\forall t \in \mathbb{N}$, $f^t(c)_j = 0$ since $\lambda(-, \ldots, -, x \mid 0, -, \ldots, -) = 0$ for $x \neq \alpha_{R_2^V}$, where $\alpha_{R_2^V}$ is the height difference coding for $R_2^V$. Remark that if $x = \alpha_{R_2^V}$ then the rule corresponding to the register $R_2^V$ has to be applied and may not return 0. Anyway, in the present case, if $j = k + 2$ then $c_{j-1}$ is a multiple of $2N + 1$ and hence $x \neq \alpha_{R_2^V}$. If $j > k + 2$ then $c_{j-1} = 0$, $x = 0 \neq \alpha_{R_2^V}$. $\square$

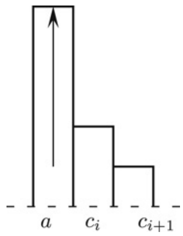



Fig. 20. The identity of $(c_i, c_{i+1})$ is not modified.

**Lemma 45.** *Consider a configuration $c \in \mathfrak{F}$. If $c$ is such that the columns $(c_i, c_{i+1})$ code for an identity $I$ then for all $t \in \mathbb{N}$, the columns $(f^t(c)_i, f^t(c)_{i+1})$ code for the same identity $I$.*

**Proof.** Let $c \in \mathfrak{F}$ be a configuration containing a symbol $I$ at position $(i, i+1)$. During a valid simulation, this pair evolves according to the local rules which may change its state or color, but preserve its identity $I$.

The only problem which could occur to change the identity of $c_i$ or $c_{i+1}$ is when a column comes "too close" on the left or on the right of the pair (see Fig. 20).

When this happens, then $c_i$ becomes an identity error and does not evolve anymore (for instance, in the Fig. 20, this happens when $0 < a - c_i \leqslant N$). To prevent $c_{i+1}$ from moving and hence maintain the identity $I$, one should just add the constraint that a local rule returns a non-zero value if and only if both members of the pair do not have ambiguity in their code. This is easy to check, for example $\lambda(\ldots, - \mid \alpha_I, x, -, \ldots) = 0$ whenever $0 < x - \alpha_I \leqslant N$ for the left column. This new constraint does not affect the simulation, as such a situation should not happen in a valid configuration. □

**Lemma 46.** *Consider a malformed configuration $c \in \mathfrak{F}$. If $\mathcal{S}_\mathcal{M}$ does not halt on $c$, then there is a lift whose color changes infinitely often.*

**Proof.** Assume $c \in \mathfrak{F}$ is malformed, and $\mathcal{S}_\mathcal{M}$ does not halt when started from $c$. Because of Lemma 44, $\forall t \in \mathbb{N}$, $|f^t(c)|$ is bounded independently from $t$. So the infinite behavior is due to "vertical" movement in $c$, *i.e.* there is a column whose content changes infinitely often. Because of the conservation of the identity shown in Lemma 45, this column is in fact a pair of columns, as its identity cannot be modified. Hence, there is a lift in $c$ which evolves infinitely often (otherwise the configuration cannot change, since pairs of columns move only when they have a lift in their neighborhood, at most once every time the lift moves).

Moreover, there are no infinite columns in configurations taken from $\mathfrak{F}$, which prevents this lift from keeping increasing or decreasing (lifts never go higher than the maximal value in their neighborhood, nor lower than the minimal one). As a consequence, its color changes infinitely often, otherwise the lift would have either stopped or gone to $\pm\infty$. Indeed, if the color does not change, the lift has no other choice but go towards the same direction after a finite number of steps. □

**Proposition 47.** *Consider a configuration $c \in \mathfrak{F}$. If $c$ contains an error (either identity or neighborhood error) then $c$ is ultimately periodic for $\mathcal{S}_\mathcal{M}$.*

**Proof.** Let $c \in \mathfrak{F}$. By contradiction, assume that $c$ contains an error (no matter if identity or neighborhood error) and is aperiodic.

First of all, Lemma 46 implies that there is a lift in the configuration, whose color changes infinitely often. Hence, there are infinitely many simulation steps **S**-**V**-**C**, which imply infinitely many correct comparison steps **C**.

In this step, $L_C$ checks the validity of all columns $C, C^V, q, q^V, R_1, R_1^V, R_2, R_2^V$. If one of them contains an error, either identity or neighborhood error, the simulation stops. This contradicts the aperiodicity of $c$. The same holds for $L_R$. It has to be valid, otherwise the next **S** step cannot be started and the simulation is blocked forever. □

Now we are ready to prove the main result of this section.

**Theorem 48.** *Both problems **ULT**($\mathfrak{P}$) and **ULT**($\mathfrak{F}$) are undecidable.*

**Proof.** First of all, remark that it is enough to prove the thesis on $\mathfrak{F}$. In fact, from any finite configuration one can obtain a periodic configuration by repeating periodically the non-zero pattern surrounded by a suitable border of zeroes (if necessary). Moreover, we provide the proof for dimension 1 only, since a similar construction can be done for other dimensions.

As said previously, we reduce these problems to the halting problem of a two registers machine with finite control started with both registers at 0.

Consider a two registers machine with finite control $\mathcal{M}$, and let $\mathcal{S}_{\mathcal{M}}$ be the associated sand automaton given by the above construction.

If $\mathcal{M}$ does not halt, then by construction there is a configuration $c \in \mathfrak{F}$ (the one coding for the input $C = C^V = 0$, $q = q^V = q_0$, $R_1 = R_1^V = 0$, $R_2 = R_2^V = 0$) which is not ultimately periodic for $\mathcal{S}_{\mathcal{M}}$. Indeed, the columns of $c$ related to the counters $C$ and $C^V$ keep increasing.

For the other implication, suppose there exists a configuration $c \in \mathfrak{F}$ such that $\mathcal{S}_{\mathcal{M}}$ is not ultimately periodic when started from $c$. By Proposition 47, $c$ has to be valid. Hence, by construction of the automaton, if $c$ is not ultimately periodic for $\mathcal{S}_{\mathcal{M}}$ then $\mathcal{M}$ does not halt when started from registers at 0 (if the computation is valid and if $\mathcal{M}$ halts then $\mathcal{S}_{\mathcal{M}}$ freezes). □

## 9. Conclusions

In this paper we introduced a new topology on $\widetilde{\mathbb{Z}}^{\mathbb{Z}^d}$ in order to have a topological "playground" for the study of sandpile-like models.

In this setting, infiniteness conserving continuous functions commuting both with the shift and the raising maps coincide with the class of sand automata (Theorem 17). We have seen that these automata are useful for generalizing sandpile models and constitute a useful formal context to study their dynamical behavior (Sections 3 and 4).

In Section 6, we investigated basic set properties such as surjectivity and injectivity. We remark that these properties are necessary conditions for many dynamical behaviors (transitivity, ergodicity and expansivity for instance). Except for simple examples, establishing if a SA is surjective (resp. injective) is a difficult task. It would be interesting to investigate the decidability of the surjectivity and injectivity properties.

In the second part of the paper we considered decidability issues about simple dynamical behavior. We proved that ultimate periodicity is undecidable by a reduction to the halting problem of a two registers machine with finite control. We believe that the proof technique might be useful for proving the undecidability of similar dynamical properties, such as nilpotency.

Another point is that we are not aware of any SA with chaotic dynamics (of course one should consider SA over the subset of configurations with neither sinks nor sources, otherwise SA are not even sensible to initial conditions, for instance). We have no examples of expansive or transitive SA.

Solving these questions would be a first step towards a classification of sand automata according to their dynamical behavior. The criteria used to distinguish the classes would have to be precise enough to characterize the behavior, but at the same time not to restrictive so that all classes contain a large number of automata.

Finally, another research direction consists in studying sand automata from a computational point of view. We wonder whether the fact that this model relies on an infinite number of states would allow unusual computations (language recognition on infinite alphabets for example), or increase the speed of what can be done by cellular automata.

## References

[1] J. Cervelle, E. Formenti, On sand automata, in: STACS 2003: 20th Annual Symposium on Theoretical Aspects of Computer Science, in: Lecture Notes in Computer Science, vol. 2607, Springer, 2003, pp. 642–653.

[2] J. Cervelle, E. Formenti, B. Masson, Basic properties for sand automata, in: Mathematical Foundations of Computer Science 2005, in: Lecture Notes in Computer Science, vol. 3618, Springer, 2005, pp. 192–211.

[3] P. Bak, C. Tang, K. Wiesenfeld, Self-organized criticality, Physical Review A 38 (1) (1988) 364–374.

[4] E. Goles, M.A. Kiwi, Game on line graphs and sandpile automata, Theoretical Computer Science 115 (1993) 321–349.

[5] E. Goles, M. Morvan, H.D. Phan, The structure of linear chip firing game and related models, Theoretical Computer Science 270 (2002) 827–841.

[6] P. Ruelle, S. Sen, Toppling distributions in one-dimensional abelian sandpiles, Journal of Physics A 25 (1992) 1257–1264.

[7] D. Dhar, P. Ruelle, S. Sen, D. Verma, Algebraic aspects of sandpile models, Journal of Physics A 28 (1995) 805–831.

[8] C. Moore, M. Nilsson, The computational complexity of sandpiles, Journal of Statistical Physics 96 (1999) 205–224.

[9] P.B. Miltersen, Two notes on the computational complexity of one-dimensional sandpiles, Tech. Rep. RS-99-3, BRICS, 1999.

[10] T. Brylawski, The lattice of integer partitions, Discrete Mathematics 6 (1973) 201–219.

[11] M. Latapy, R. Mantaci, M. Morvan, H.D. Phan, Structure of some sand piles models, Theoretical Computer Science 262 (2001) 525–556.

[12] M. Latapy, H.D. Phan, The lattice structure of chip firing games and related models, Physica D 155 (2001) 69–82.

[13] E. Goles, M. Morvan, H.D. Phan, Sandpiles and order structure of integer partitions, Discrete Applied Mathematics 117 (1–3) (2002) 51–64.

[14] E. Goles, M. Latapy, C. Magnien, M. Morvan, H.D. Phan, Sandpile models and lattices: A comprehensive survey, Theoretical Computer Science 322 (2004) 383–407.

[15] B. Durand, Global properties of cellular automata, in: Cellular Automata and Complex Systems, Kluwer, 1998.

[16] G.A. Hedlund, Endomorphisms and automorphisms of the shift dynamical system, Mathematical Systems Theory 3 (1969) 320–375.

[17] P. Kůrka, Topological and symbolic dynamics, in: Undergraduate Texts, vol. 11, Société Mathématique de France, Paris, 2003.

[18] K. Nagel, M. Schreckenberg, A cellular automaton model for freeway traffic, Journal de Physique I 2 (1992) 2221–2229.

[19] B. Durand, E. Formenti, Z. Róka, Number conserving cellular automata I: Decidability, Theoretical Computer Science 299 (2003) 523–535.

[20] E. Formenti, A. Grange, Number conserving cellular automata II: Dynamics, Theoretical Computer Science 304 (2003) 269–290.